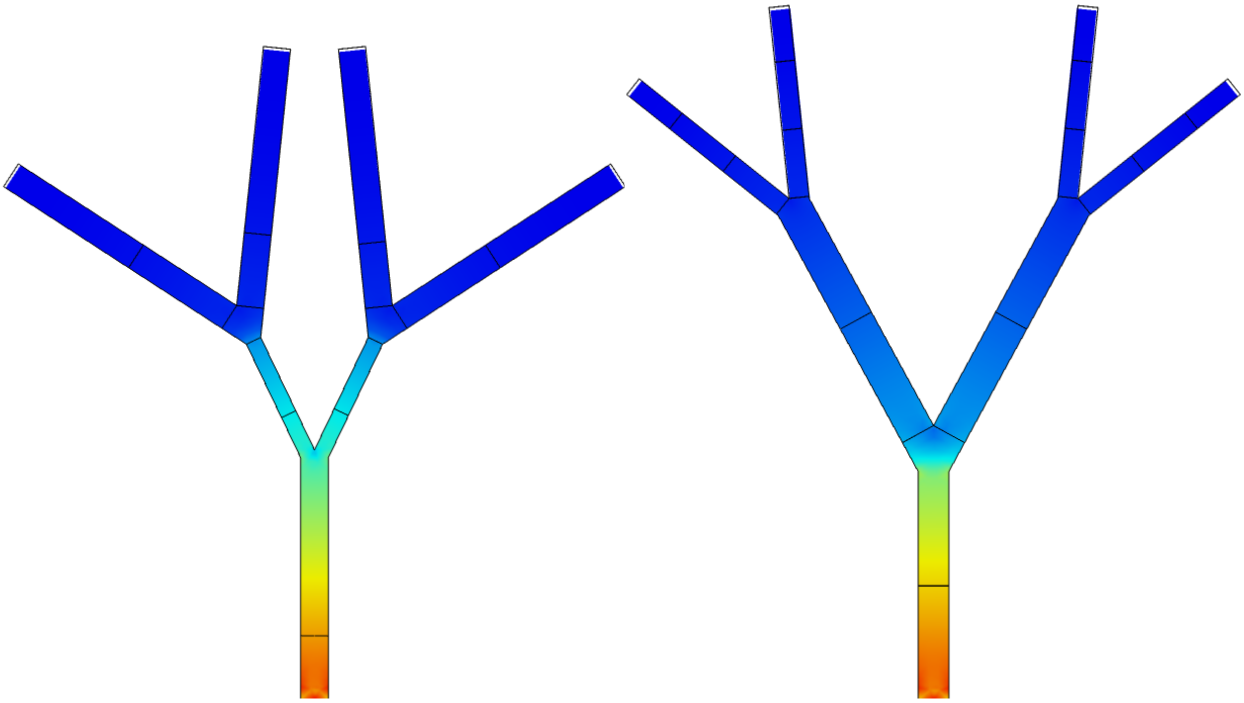
CHALMERS
UNIVERSITY OF TECHNOLOGY



Design and Optimization of Nature-inspired Piezoelectric Energy Generators: Fractal design

Master’s Thesis in Applied Mechanics 2021:23

BOGDAN PAMFIL
RICHARD PALM

DEPARTMENT OF MECHANICS AND MARITIME SCIENCES

CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden 2021
www.chalmers.se

Master's Thesis in Applied Mechanics 2021:23

Design and Optimization of Nature-inspired Piezoelectric Energy Generators: Fractal design

BOGDAN PAMFIL
RICHARD PALM

Department of Mechanics and Maritime Sciences
Division of Dynamics
Chalmers University of Technology
Gothenburg, Sweden 2021

Design and Optimization of Nature-inspired Piezoelectric Generators: Fractal design
BOGDAN PAMFIL
RICHARD PALM

Supervisor: Agin Vyas,
Chalmers Department Microtechnology and Nanoscience, Electronics Material and Systems Laboratory

Examiner and supervisor: Peter Folkow,
Chalmers Department of Mechanics and Maritime Sciences, Division of Dynamics,
Head of Division Dynamics

Master's Thesis in Applied Mechanics 2021:23
Department of Mechanics and Maritime Sciences
Division of Dynamics
Chalmers University of Technology
SE-412 96 Gothenburg
Telephone +46 31 772 1000

Cover: Static stress distribution for two different fractal energy harvester designs generated by simulations in COMSOL Multyphysics.

Printed by Chalmers Reproservice
Gothenburg, Sweden 2021

Design and Optimization of Nature-inspired Piezoelectric Energy Generators:
Fractal design
BOGDAN PAMFIL, RICHARD PALM
Department of Mechanics and Maritime Sciences
Division of Dynamics
Chalmers University of Technology

Abstract

A piezoelectric energy harvester or generator is a device with no need for maintenance or external energy being provided since it utilizes vibrations generated by industrial machines to produce an electrical voltage output through a piezoelectric material. The piezoelectric energy harvester can be used to power small Internet of Things (IoT) components, without need for an external battery or connecting cables. The optimization solutions are meant to develop a proof-of-concept for the chosen fractal tree energy harvester design. The aim of this is to investigate if that design can produce a sufficiently high electric output (voltage and power) and a high enough stress in longitudinal branch direction as Frequency Response Functions (FRFs).

The chosen material for the overall structure is structural steel, whereas the piezoelectric material for energy harvesting is Polyvinylidene Fluoride (PVDF). The fractal tree design is optimized by running a MATLAB code called SAMO which carries out first a Sensitivity Analysis (SA) followed by a Multi-objective Optimization (MO) using an Elitist Genetic Algorithm (GA). The coupling between MATLAB and the COMSOL Multyphysics fractal tree model is ensured by the COMSOL feature called LiveLink for MATLAB. The optimal design solutions form a set which are referred to as a Pareto set, and they are associated to two minimized objective functions and multiple design variables. In the first phase of the optimization setup (Phase 1), for the minimization of two given objective functions per optimization process, multiple fractal tree geometry design variables are tested. For that first phase, at each optimization procedure iteration, the overall geometry of the fractal design is updated according to the design variables values, and if the geometry is acceptable, a static analysis is computed. Then, if the maximal static load is permissible, a modal analysis (eigenfrequency study) and a frequency domain analysis are also carried out. For the second optimization phase (Phase 2) of the project, two general designs of the fractal tree are chosen, one with and one without proof masses (PM) positioned at the end of the top branches. The piezoelectric material placement is optimized by considering an area coverage variation for each branch of the fractal tree design in order to favour the least negative voltage output generation and maximal branch longitudinal stress in the frequency domain.

In the end, the best optimized fractal tree design in terms of calculated longitudinal stress and voltage output will be fabricated after completion of the project. Simulated FRFs responses will then be validated in the MC2 Laboratory by being compared with experimental FRFs obtained by Laser Doppler Vibrometer tests.

Keywords: Energy harvester, piezoelectric, bandwidth, fractal design, Genetic Algorithm, Pareto set, objective function, Frequency Response Function.

Acknowledgements

First of all, we would like to thank Associate Professor Peter Folkow for proposing to us this interesting project which has very promising technological prospects. He has given us his thoughtful insight to help us make decisions along the way and he made us feel confident that we were going to reach our goals.

Agin Vyas, as a PhD student from Chalmers' Department of Microtechnology and Nanoscience, also regularly guided us and dedicated his time to help us with setting up the FE models and take into account geometrical and computational considerations. We are very thankful for all the advice and ideas that he has have given us along the project.

Other collaborators from the Department of Microtechnology and Nanoscience include Professor Peter Enoksson and Associate Professor Per Lundgren who gave us great technical support and recommendations during meetings.

We would also want to thank Professor Viktor Berbyuk for sharing with us and giving us the opportunity to use the SAMO code for the completion of this project. We are very grateful for this, since the project otherwise would not have been as successful as it has become.

Furthermore we are also appreciative for RISE Smart hardware's close involvement in this project, namely Adjunct Professor Cristina Rusu's and Doctor Henrik Staaf's collaboration. We feel privileged to have worked with RISE since the future experimental characterization and analysis of our fractal energy harvester design will be carried out within their facilities.

Finally, Johan Andersson, a former Chalmers student who conducted research also on the improvement of piezoelectric energy harvester characteristics, provided us with his experience and knowledge on this topic. His word of advice helped us start the project and clarify some doubts.

"Il y a deux sortes d'esprit: l'esprit de géométrie et l'esprit de finesse." - Blaise Pascal

Bogdan Pamfil and Richard Palm, Gothenburg, June 2021

Nomenclature

Abbreviations	
BC	Boundary condition
Bnd	Bound
BW	Bandwidth
CF	Voltage over Stress Conversion Factor
COV	Covariance
Dist	Distribution
eig.freq.	Eigenfrequency
FE	Finite Element
freq. rng.	Frequency range
FRF	Frequency Response Function
GA	Genetic Algorithm
Gen	Number of generations
GSA	Global Sensitivity Analysis
IoT	Internet of things
Iter.	Iteration
lb	Lower bound proportion limit
MBSD	Multi-Body System Dynamics
MO	Multi-objective Optimization
No.	Number
Not.	Notation
OF1, OF2	Objective function 1, objective function 2
OFC	Objective function condition
PM	Proof Masses
Pop	Population size
Pts	Points
PVDF	Polyvinylidene Fluoride or Polyvinylidene Difluoride
PZT	Lead Zirconate Titanate
SA	Sensitivity Analysis
SAMO	Sensitivity Analysis and Multi-objective Optimization
SF	Security Factor
sol.	Solution
TH	Stress FRF Threshold
TH_V	Voltage FRF Threshold
ub	Upper bound proportion limit
vM	von Mises stress

Greek Letters

α	Angle between mid branches
α_3	Angle between top branches
α_{dM}	Rayleigh damping factor associated to Mass matrix
β_{dK}	Rayleigh damping factor associated to Stiffness matrix
Δ	Previous fractal tree design length reduction factor of upper branches
ω	Frequency limit for the frequency domain analysis in radians
Σ	Sum of a specific variable for each i^{th} branch, such as surface averaged stress or voltage
σ	Branch surface averaged stress
σ_n	Branch surface averaged stress in the longitudinal direction of the branch
σ_x	Surface averaged normal stress in the x direction of the global coordinate system
σ_y	Surface averaged normal stress in the y direction of the global coordinate system
$\sigma_{n,i}$	i^{th} branch surface averaged stress in the longitudinal direction of the branch
τ_n	Branch surface averaged shear stress with respect to the stress in longitudinal direction of the branch
τ_{xy}	Surface averaged shear stress in the y direction of the global coordinate system with x direction being perpendicular to the shear plane
θ	Angle of each fractal tree branch with respect to x direction in global coordinate system
ζ_{d1}	Rayleigh damping damping ratio associated to first reference eigenmode
ζ_{d2}	Rayleigh damping damping ratio associated to second reference eigenmode

Roman Letters

$\overline{a}$	Amplitude of the acceleration harmonic excitation in the time domain
$\overline{d}$	Amplitude of the displacement harmonic excitation in the time domain
$\overline{v}$	Amplitude of the velocity harmonic excitation in the time domain
A	Area
A_i	Area of i^{th} branch
f	Frequency
f_i	i^{th} Eigenfrequency
f_{d1}	Rayleigh damping first reference eigenfrequency
f_{d2}	Rayleigh damping second reference eigenfrequency
f_{limit}	Frequency limit for the frequency domain analysis in Hz
L	Length of trunk branch, 1^{st} branch length
L_2	Length of branches for the first level, mid branches length (2^{nd} and 3^{rd})
L_3	Length of branches for the second level, top branches length (4^{th} to 7^{th})
L_i	Length of i^{th} branch
w	Width of trunk branch, 1^{st} branch width
w_2	Width of branches for the first level, mid branches width (2^{nd} and 3^{rd})
w_3	Width of branches for the second level, top branches width (4^{th} to 7^{th})
$\mathbf{C}$	FE model Damping matrix
$\mathbf{K}$	FE model Stiffness matrix
$\mathbf{M}$	FE model Mass matrix

a(t)	Acceleration harmonic excitation in the time domain
d(t)	Distance harmonic excitation in the time domain
g	Gravity of Earth constant
N	Number of branch levels, number of fractal tree branch extensions
V	Voltage
v(t)	Velocity harmonic excitation in the time domain

Contents

List of Figures

List of Tables

1 Introduction

In the near future more and more microsystems, such as sensors, will be integrated in different applications. These microsystems will be connected to other systems and be part of the Internet of Things (IoT), which requires them to have an energy source. Conventionally, energy can be transferred through electric cables from an external energy source, or by using a battery. However, it would be more convenient if energy could be provided in other ways that do not require maintenance or added complexity in the manufacturing process. Having a device that permanently provides energy is therefore very desirable. A piezoelectric energy harvester utilizes the vibrations that are induced from industrial machines or other technological applications, by achieving an electrical voltage output from piezoelectric materials that become strained due to vibrations.

Piezoelectric energy harvesters are promoted as renewable energy solutions which have great potential for many technological applications by being fixed to a hosting structure in motion. The hosting structure can be for example a building or a bridge, a vehicle vibrating component, or even a bio-mechanical device [1].

The chosen fractal design concept is a piezoelectric energy harvester which can be classified as a stress distribution optimized structure. Another example of piezoelectric energy harvester pertaining to this category would be a trapezoidal shaped cantilever beam. Unlike an uniform beam, the trapezoidal shaped one will lead to an increased life expectancy (high endurance) of the structure and a higher energy efficiency (high power generation) due to less stress concentrations in some regions [2].

1.1 Background

Previous work and studies regarding this subject have been done by Vyas [3] and Andersson [4], where two-degrees-of-freedom M-shaped cantilever micro-energy harvesters have been studied. One of the main issue is to have an electric voltage output that is high enough for a broader frequency range. Usually the voltage output which is useful for energy harvesting, is only high enough at eigenfrequencies or in the vicinity of them. Since the vibrations of the source might vary, it is desirable to design an energy harvester that can generate energy for a broader frequency range. Andersson [4] showed that the shape of a two-degrees-of-freedom micro-energy harvester can be designed so that eigenfreqencies are usable for power extraction and closer to each other. The voltage outputs, which are particularly high at eigenfrequencies and generated by individual beams, can be summed to produce a total voltage output. This total voltage is the result of individual beams voltage outputs which are overlapping each other and produce a harvestable energy. This overlapping of voltage Frequency Response Functions (FRFs) due to eigenfrequencies being close to each other can be observed in Figure 1.1.

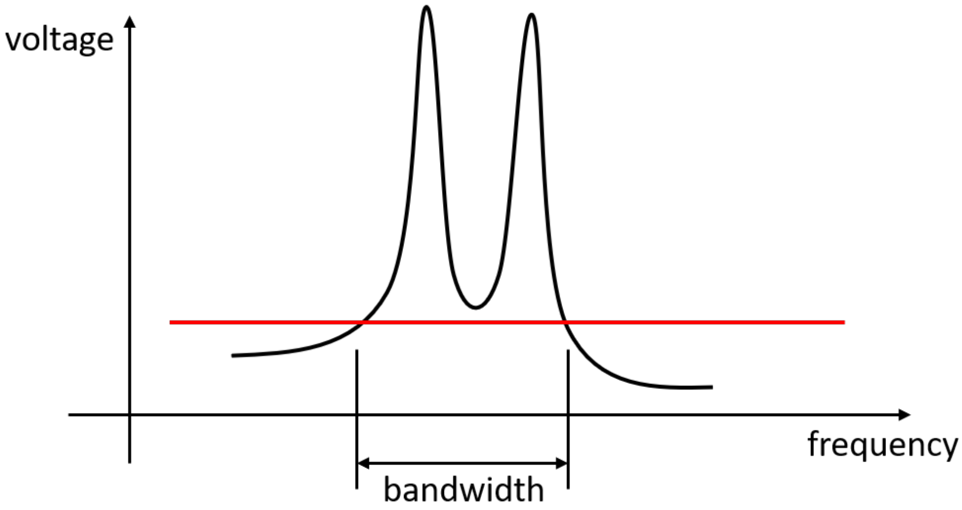


Figure 1.1: Voltage FRF with voltage threshold in red. Two modes' eigenfrequencies are close enough to each other so as to produce a larger bandwidth.

Another interesting idea for a new energy harvester design which could be useful for future applications, is the so called fractal design. This chosen design has a cantilever beam with one of its end being clamped to a vibrating source. Starting from the free end of that cantilever beam, two other beams extend as two branches with a defined angle between them, and this altogether forms a Y-shape structure. The two extending branches can then continue to have two more branches extending from their respective free ends, and this increases the number of branches for the whole structure. The fractal energy harvester concept design can be seen in Figure 1.2. Naturally, this design can have different configurations, e.g. different angles between branches, different lengths of branches or different widths of branches. Depending on the overall design, there can be more usable eigenmodes other than the pure bending one, such as torsional mode shapes, that can contribute to produce a higher total voltage output.

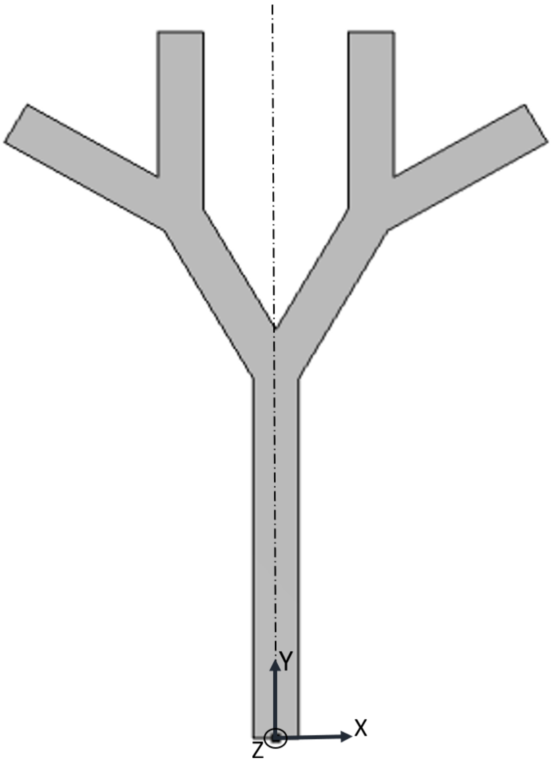


Figure 1.2: Fractal energy harvester geometry design example generated in COMSOL Multyphysics with two level of branches. The point at the bottom edge is the trunk root of the fractal tree and it is located at the origin of the global coordinate system, meaning at $x = 0$, $y = 0$ and $z = 0$ (0,0,0).

Several concept designs for the piezoelectric energy harvester, other than this chosen fractal tree design, have been investigated prior to the start of this project. For all previous designs and the current one, the main idea is to use a vibrating beam where strained piezoelectric material layers placed onto the structure are used to maximize usable energy production. The electric output (voltage and power) is usually too low in the lower frequency domain when the vibrating excitation frequency is not close to an eigenfrequency. The cantilever beam concept design (see Figure 1.3) is a similar basic design to which can be added a proof mass at its end so that the voltage FRF output is increased and the structure's eigenfrequencies are reduced. Those characteristics are desirable for an energy harvester.

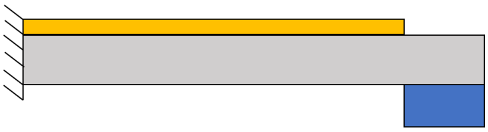


Figure 1.3: Cantilever beam concept design with piezoelectric material in yellow and proof mass in blue.

Setting up a procedure to optimize a fractal energy harvester design can be done in many ways. A MATLAB code named `SAMO.m`, coded by Seyed Milad Mousavi Bideleh and Viktor Berbyuk at the Department of Mechanical and Maritime Sciences M2, Chalmers University of Technology, Gothenburg, Sweden, uses an Elistist Genetic Algorithm (GA) to perform multi-objective optimizations [5]. The GA is useful for optimization problems that are discrete or nonlinear [6], and `SAMO.m` can therefore be considered as a good tool to optimize a fractal energy harvester design. The GA (evolutionary method) is also one solution technique amongst other guided random search methods that have the benefits of finding optimal solutions for objective functions which are allowed to be discontinuous or non differentiable. On the other hand, classical optimization techniques are not well adapted to multi-objective functions. They commonly rely on the gradient to locate a global optimum (either maximum or minimum) and thus they are more prone to capture a local rather than global optimum solution [7].

The MATLAB multi-objective function `gamultiobj.m` is used in `SAMO.m`, and to understand better that function one can refer to MATLAB's documentation. It could also be very useful to have a look at MATLAB's documentation related to the Global Optimization Toolbox [8] and also at the documentation related to the single objective optimization function `ga.m` [9] to understand the general reasoning behind the GA.

The work performed by Andersson [4] was mainly done using the commercial Finite Element (FE) software COMSOL Multiphysics. His work made use particularly of the COMSOL Composite Materials Module for voltage FRF computation due to stress applied on the piezoelectric material layers which are deposited on the energy harvester's structure. Besides, MATLAB can be coupled with COMSOL through the LiveLink for MATLAB feature [10]. The coupling between COMSOL and MATLAB via LiveLink has already been done according to a paper published in the Applied Mechanics and Materials periodical which treats of the optimization of underground thermal processes [11]. Appendix L clarifies the setup for creating the FE models in COMSOL and for exporting simulations results.

1.2 Project aim

The aim of this project is to optimize a fractal energy harvester design for future industrial IoT (Internet of Things) applications. The design should be able to have an electric voltage output of 100 mV and above for a broad frequency range being as large as possible. The design should also be able to have a considerable voltage output for frequencies up to 200 Hz and with an acceleration going up to 2 g-force. The fractal energy harvester has to withstand sudden impacts, e.g. if it is dropped when being handled, by not breaking for static accelerations going up to 10 g-force.

1.3 Limitations

There is a limitation in terms of size for the fractal energy harvester design. In this project the design will be of macro-scale since a prototype will be manufactured for future validation. In this case, the geometry of the fractal energy harvester cannot be outside set boundaries of 8 cm × 8 cm, due to the manufacturing process (laser cutting).

The main material of the fractal energy harvester structure that is used for simulations is structural steel. The simulations for this project will be carried out using the FE software COMSOL Multiphysics, where the default structural steel from the software's material library is selected. The two main piezoelectric materials which can be considered for an energy harvester are the Polyvinylidene Fluoride (PVDF) and the Lead Zirconate Titanate (PZT). Only the PVDF material is used to collect data regarding the voltage output. Even though the PZT has better piezoelectric properties than the PVDF, the PVDF has a much better flexibility, a low density, an easy availability and does not contain any toxic constituent like lead within the PZT [2].

The steel material thickness is set at 0.2 mm and the PVDF material one is 0.028 mm. Other material thicknesses were not implemented within the FE models, because only these material thicknesses are provided by the manufacturers, and it would be much harder to fabricate a design with different thicknesses.

The fractal tree design will only have two levels of branches extending from the trunk bottom branch, i.e. 7 branches in total.

2
Theory

The optimization of the fractal energy harvester design is executed by relying on the `SAMO.m` code. The acronym SAMO stands for Sensitivity Analysis (SA) and Multi-objective Optimization (MO), and for the optimization application of this project, it is not a necessity to do a Sensitivity Analysis. Nonetheless, it can be of interest to analyze which design variables are most influential for any parts of the optimization procedures.

The optimal solutions provided by the `SAMO.m` code are given as a Pareto set and Pareto front. The Pareto set represents the design configurations with a specific set of design variables which contribute to the minimization of objective functions whose values are forming the Pareto front.

The FE model of the fractal energy harvester is built in COMSOL and needs damping properties which are configured according to the Rayleigh damping theory. The FRFs results chosen for optimization are stress FRFs rather than voltage, since computational time needs to be reduced, and for them to be computed in COMSOL the dynamic load case needs to be specified. For the FRFs results to be usable for the optimization purpose, the known voltage threshold must be converted to a corresponding stress threshold.

With the above in mind, there are some theory concepts which are necessary to be understood in order to use consciously the `SAMO.m` code and set up properly the fractal energy harvester FE models. Those are presented in the following sections.

2.1 Sensitivity analysis

In a sensitivity analysis the input parameters to a system or function are varied individually to see how an output is affected. If an individual input parameter is varied and it drastically changes the outcome of the output value, it can be said to be highly sensitive. Instead, when the outcome of the output is only changed slightly, the sensitivity of a parameter is said to be low.

Performing a sensitivity analysis can be very useful for an optimization problem, since it can indicate which parameters have the most influence on objective functions values. Hence, the sensitivity analysis can be relevant for an optimization problem which is computationally heavy and simulation time needs to be reduced. Input parameters with low sensitivity can be excluded from being varied during the optimization procedures and those can be set to have constant nominal values instead. However, the outcome of the sensitivity analysis for each input parameter also depends on the number of different values that are taken into account for its estimation. Additional explanations about the sensitivity analysis (SA) settings in SAMO are given in Appendix B.

2.2 Multi-objective optimization

When there are more than a single objective function to optimize, a problem can be defined as a multi-objective optimization problem. Adding more functions to optimize simultaneously will often not give a single optimal solution, but instead a set of different optimal solutions. Depending on the problem at hand, a single optimal solution needs to be carefully selected based on the set of solutions (Pareto front and set) and by weighing in or prioritizing the objective function which is more important than the others. The complexity of finding good optimal solutions increases when the number of objective functions that are optimized simultaneously is increased. In practise, it is harder to analyse more than three objective functions simultaneously, since it then becomes more difficult to visualize the Pareto front and make the selection of one design amongst the set of optimal solutions [12].

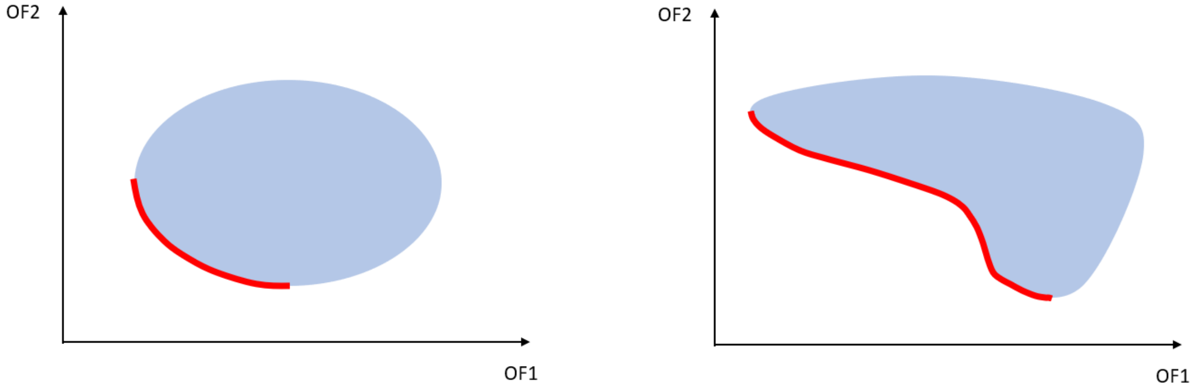
2.2.1 Pareto set and front

When the objective functions are defined for an optimization problem, the results can be plotted on a graph as a Pareto front. For example, when having two objective functions, the results can be presented in a 2D plot, where the x-axis shows the values of the first objective function (OF1) and the y-axis shows the values of the second one (OF2). Each single point in the graph represents an optimal objective function value which is linked to a given set of design parameters or variables. Moreover, the Pareto set represents the set of design variables which is linked to a specific optimal solution from the Pareto front.

If an optimization problem requires that one objective function is minimized while the other one is maximized, it can become unclear from a Pareto front plot what would be a good optimal solution to be selected. To counter this issue, it would be more practical to redefine the objective functions so that both of them are either minimized or maximized, which will provide a clear plot of the optimal solutions, see example in Figure 2.1. An objective function which needs to be maximized can become a minimized objective function by being defined as one divided by the maximized value, i.e. $OF_{min} = 1/OF_{max}$.

In Figure 2.1 two examples are shown of optimal solutions for two objective functions. The shaded area represents the totality of feasible solutions that are included in an optimization procedure. The red lines represent what would be the set of minimized optimal solutions for an optimization procedure, and those are known as Pareto front. Note that since there are two objective functions in this case, there is no single solution that dominates over all others. Each point of the red lines are optimal solutions where both the first and the second objective functions are minimized simultaneously. Any point to the right or above the red lines has either a higher value for the first or for the second objective function [13].

Based on the shape of the Pareto front, one can analyse the trade-offs and pick one or more options from the curve. For some cases this is easier said than done. For instance, in Figure 5.2b the shape of the Pareto front would be more difficult to analyse and a large trade-off between favouring more OF1 or OF2 might be considered. In Figure 5.2a one could say that an optimal solution choice is more clear and that an option would probably be picked around the middle of the red line. One must of course first analyse the actual values of OF1 and OF2 before making a decision.



(a) (b)

Figure 2.1: Two examples of Pareto fronts (red lines) which show the minimized values for two objective functions OF1 and OF2. The shaded areas show the complete set of feasible solutions.

2.2.2 Genetic Algorithm

The Genetic Algorithm (GA) is inspired by how natural selection works in biological evolution, and it is very convenient for problems that do not work well with classic optimization, e.g. discontinuous or extremely non linear problems [6]. It uses the principle of probabilistic searching to start converging towards the solution and creates a population of individuals with different combinations of design parameters. Each individual is tested to see how it performs according to the defined objective functions. The individuals that perform best will then become parents and will influence how the next population generation of different design parameters will be generated. Depending on how well the initial guess of design parameters and other input settings are, the GA will eventually find optimal solutions that have converged [5].

The GA converges very well towards the Pareto front of global solutions by relying on an initial guess of solution [14]. To update the solution consisting of a set of design variables, the Elitist GA applies the following five steps to shape the Pareto front: Chromosome encoding, fitness, selection, recombination and then evolution [15]. To ensure a good convergence of the Pareto set of design variables results, it is important to carefully select a good combination of the GA's settings, including the number of generations and the population size. The maximum number of iterations, performed by the SAMO optimization procedure, is the result of the product of the population size with the number of generations. The SAMO code carries out optimization iterations until a convergence is reached in terms of a satisfying Pareto front, or otherwise until the maximum number of of iterations is reached [5].

For this project's application there are two objective functions intended to be minimized per optimization procedure, and there is no weighting coefficient which is applied to the objective functions vector (2 main objective functions, and one condition objective function). With no weighting coefficient applied to the objective functions vector, the GA attributes an equal importance to minimizing all objective functions. Consequently there is no use for using a normalization factor which multiplies the objective functions.

The SAMO user guide [5] gives a short introduction about the main steps of the GA which are presented in the following sections. For more detailed explanations one can also refer to [7].

2.2.2.1 Chromosome encoding

The GA starts by selecting values of each design parameter for each individual that forms the population. Each design parameter is usually encoded as a binary string number, and forms what is called a gene. All of the design parameters, or genes, of an individual will then together form a longer binary code which is known as a chromosome. The chromosome is thus an exclusive code or explanation of a single individual, and it is the GA's way of creating and managing an individual of the population. The gene will, when the GA decodes it, give a value in the interval between the lower and upper bound limits that are set to a design parameter. Note that the size of the binary string that the GA decodes will affect the precision of the design parameter value.

2.2.2.2 Fitness

When the GA has decoded a chromosome to the real values of each design parameter of an individual, it will test the individual's performance. The performance, also known as the fitness, decides if the individual is better or worse than the other individuals of the population. The fitness first needs to be defined, either as one or a collection of objective functions, or a certain combination of them. When all chromosomes have been tested and given a fitness value, the GA can proceed to the next generation of individuals, which will become influenced by the chromosomes that have the best fitness.

2.2.2.3 Selection

In the selection process, the chromosomes that are suitable parents for the recombination of new individuals for the next generation are being identified. There exists different methods for the selection process, and two common ones are Roulette wheel and Tournament. With the Roulette wheel method a chromosome's chance of being selected is proportional to the fitness value. Once a chromosome has been picked, it will not be excluded for the the next parent selection, and the same chromosome can be chosen several times for recombination of individuals of the next generation. In the Tournament method, a pair or small group of chromosomes are first randomly selected from the population. Their fitness values are compared amongst them, and the one with the best fitness value will become a parent to a new individual of the new generation. This is then repeated for every new individual of the next generation.

In parallel to this, an Elite count can be set. This number ensures that some of the individuals will survive and be part of the next generation with no recombination method (crossover or mutation) modifying them.

2.2.2.4 Recombination

The next step of the GA is the Recombination. This is when new chromosomes for the next generation are created and they are based on selected parent chromosomes from the previous step. Two main methods apply for this, Crossover and Mutation. With Crossover, two chromosome binary strings from the selection process are first chosen. One random point (or more) along the string will then split the binary codes, and the cut parts are then carried over to each other, see Figure 2.2. New chromosome codes have then been created, with the idea that they now are a mix between the parents that were selected. To increase the diversity of the new population, Mutation is applied to a few of the new chromosomes. The Mutation operation has a small probability and switches only few of the ones to zeros and vice versa.

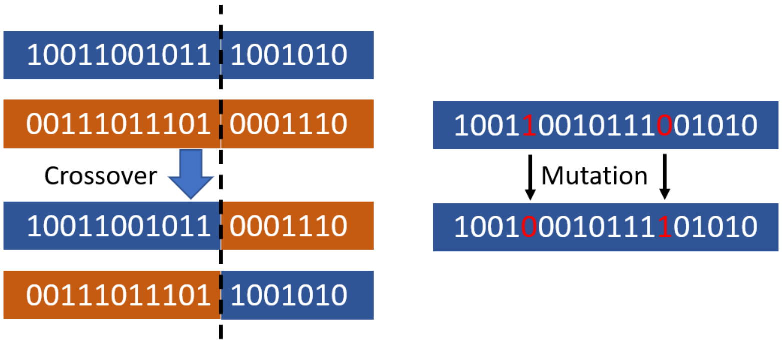


Figure 2.2: Two main methods of Recombination that are part of the GA.

2.2.2.5 Evolution

Once the selection and recombination processes are done, a new complete population has been created for the next generation. The GA will then repeat the process with fitness, selection and recombination until convergence of the fitness values is achieved, or until the maximum number of iterations is reached.

2.3 Rayleigh damping

Rayleigh damping is a linear and proportional viscous type of damping which implies modal damping on all modes. The damping matrix **C** is defined as a linear combination of mass **M** and stiffness **K** matrices, with associated scalar factors α_{dM} and β_{dK}. These two weighting scalar factors can also be determined by using two modes' eigenfrequencies and damping ratios.

The mass **M** and stiffness **K** matrices are obtained from the FE discretization of each node's inertia (mass) and stiffness using shell elements for this project. The nodal mass and stiffness have for shell elements corresponding values related to the six degrees of freedom, three translations and three rotations.

Rayleigh damping needs not only two provided eigenfrequencies in Hz (f_{d1} and f_{d2}), but also corresponding damping ratios for each one of the two modes (ζ_{d1} and ζ_{d2}) [16]. These two damping ratios associated to their corresponding mode are given a commonly chosen value of 0.001 ($\zeta_{d1} = \zeta_{d2} = 0.001$) for this kind of application.

The following equations show how the Rayleigh damping is computed.

$$\begin{aligned}
\mathbf{C} &= \alpha_{dM}\mathbf{M} + \beta_{dK}\mathbf{K} \\
\alpha_{dM} &= 4\pi f_{d1} f_{d2} \frac{\zeta_{d1} f_{d2} - \zeta_{d2} f_{d1}}{f_{d2}^2 - f_{d1}^2} \\
\beta_{dK} &= \frac{\zeta_{d2} f_{d2} - \zeta_{d1} f_{d1}}{\pi(f_{d2}^2 - f_{d1}^2)}
\end{aligned}$$

2.4 Dynamic load case

The dynamic excitation is applied in the out of plane direction along the edge of the base of the fractal tree, see Figure 2.3. Its amplitude is a 2 g-force with respect to the maximal frequency of interest considered in the FRF analysis. That maximal frequency or frequency limit is 200 Hz, since the project's mandate is to optimize the device's energy production up until that frequency. The dynamic load case considers a harmonic excitation acceleration amplitude ($\overline{a} = 2g$), and the goal is to find the equivalent amplitude for the displacement harmonic excitation ($\overline{d}$). The amplitude sign difference (opposite) between acceleration and displacement excitation load cases is ignored here because it does not change results at all. The displacement harmonic excitation amplitude ($\overline{d}$) needs to be computed so that it can be used for the frequency domain analysis dynamic load case in COMSOL. The displacement harmonic excitation amplitude ($\overline{d}$) is inverse proportional to the power of two to the frequency limit in radians (ω).

The following equations display the computational steps required to do in order to obtain the displacement harmonic excitation amplitude for the dynamic load case.

$$\begin{aligned}
f_{limit} &= 200\text{Hz}, g = 9.81\text{m/s}^2, \overline{a} = 2g \\
\omega &= 2\pi f_{limit} \\
d(t) &= \overline{d} \cdot \sin(\omega t) \\
v(t) &= \overline{v} \cdot \cos(\omega t) = \overline{d} \cdot \omega \cdot \cos(\omega t) \\
a(t) &= \overline{a} \cdot \sin(\omega t) = \overline{d} \cdot \omega^2 \cdot \sin(\omega t) \\
\overline{d} &= \frac{2g}{\omega^2}
\end{aligned}$$

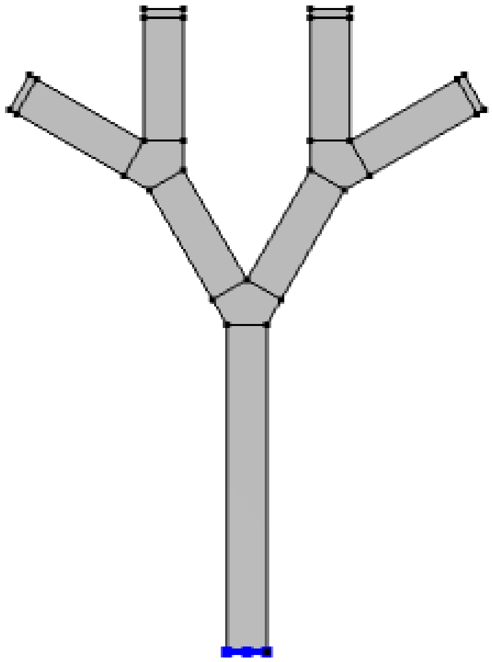


Figure 2.3: The blue colored line along the bottom edge of the fractal energy harvester geometry indicates where a prescribed displacement is set for the dynamic load case. For the static load case it is a prescribed null displacement and null rotation which are set along the same bottom edge.

2.5 2D plane stress - normal stress in the longitudinal direction of a branch

Only the normal stress in the longitudinal direction of each branch (σ_n) is of interest for the stress FRF maximization optimization of FRF characteristics, whereas the shear stress (τ_n) is of less interest. The angle θ represents the angle of each branch in the fractal energy harvester's case, with respect to the x direction of the global coordinate system. Thus, with each branch's angle known, the surface averaged longitudinal stress can be computed for each branch section using the following stress tensor transformation equations.

Figure 2.4 shows the convention of the stress components in 2D that these equations apply for.

$$\begin{aligned}
\tau_{xy} &= \tau_{yx} \\
\sigma_n &= \frac{\sigma_x + \sigma_y}{2} + \frac{\sigma_x - \sigma_y}{2}\cos(2\theta) + \tau_{xy}\sin(2\theta) \\
\tau_n &= -\frac{\sigma_x - \sigma_y}{2}\sin(2\theta) + \tau_{xy}\cos(2\theta)
\end{aligned}$$

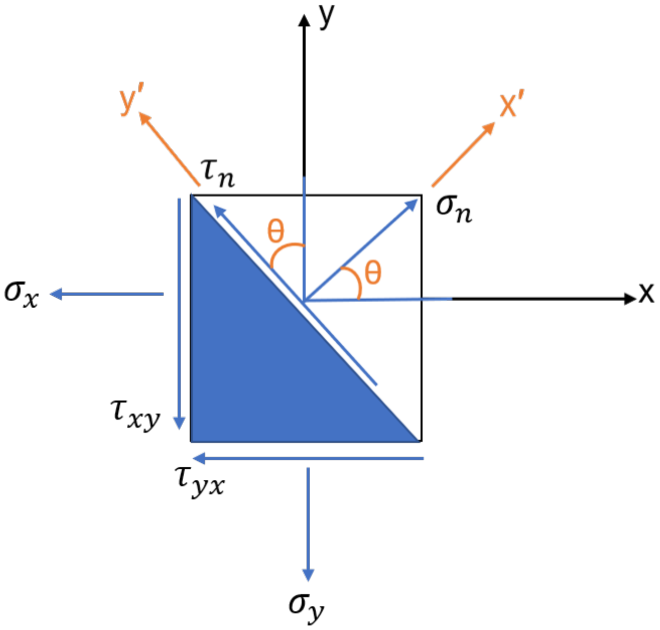


Figure 2.4: Stress tensor transformation from the global coordinate system x-y to the local coordinate system x'-y'.

2.6 Conversion factor voltage over stress

The voltage FRF values and FRF threshold of 100 mV can be both respectively converted to stress FRF values and to a stress FRF threshold. For this project, it is of interest to convert specifically the voltage FRF threshold (TH_V) of 100 mV to a stress threshold (TH). To do so, an approximated conversion factor of voltage over stress (CF) is computed. More details about how this is done are explained in the subsection 3.1.3. That conversion factor (CF) is the quotient of voltage over longitudinal stress (σ_n) for each FRF frequency sample. The scalar value of the conversion factor is taken as the mean value of the conversion factor FRF.

An example is shown below of how a stress FRF threshold can be computed considering a certain scalar conversion factor and the known voltage FRF threshold.

$$\mathrm{TH}_V = 100\mathrm{mV}$$

$$\mathrm{CF} \approx 5.58 \cdot 10^{-8} \frac{\mathrm{V}}{\mathrm{Pa}}$$

$$\mathrm{TH} = \frac{\mathrm{TH}_V}{\mathrm{CF}}$$

$$\mathrm{TH} \approx 1.79 \cdot 10^{6}\mathrm{Pa}$$

3
Methodology

In the first section of the *Methodology* chapter, the overall setups of the optimization procedures are summarised. The sections that follow are describing more in details how the Phases optimization setups are built. Following this, flow charts are presented to visualize how the main code SAMO is organized and how the main processes of all Phases optimization setups connect to each other.

3.1 Summary of the optimization procedures

The fractal energy harvester design optimization is divided into two main Phases optimization setups called Phase 1 and Phase 2. In Phase 1 the overall geometry design of the fractal energy harvester is varied and optimized for. In Phase 2 the piezoelectric material area coverage is varied to fine-tune the final outputs of the fractal energy harvester. In Phase 0, preceding both other Phases optimization setups, two eigenfrequencies (1^{st} and 5^{th}) are minimized so that a preliminary design solution can be chosen. The preliminary design's dimensions are used as Genetic Algorithm nominal values for the Phase 1 optimization setup. The 1^{st} and 5^{th} eigenfrequencies are good choices for minimization optimization because usually they fit within the frequency range of 0 to 200 Hz. This means that the 5^{th} eigenfrequency is usually, for most fractal tree geometries, the last eigenfrequency below 200 Hz (frequency limit). Phase 1 consists of optimizing the fractal tree geometry by maximizing the cumulative surface averaged longitudinal stress. For Phase 1, there are two design options, one considers the placement of proof masses at the end of the top branches, and another design option does not consider that. Finally, in Phase 2, the purpose of the optimization setup is to maximize voltage and power output by maximizing also the surface averaged cumulative longitudinal stress with the consideration of the piezoelectric area coverage variation. In Appendix F, additional explanations are provided about the chosen objective functions for all optimization Phases and the reason why those are picked.

For all Phases there are two main MATLAB codes, `SAMO.m` and `MBSD.m` (self-coded), which together run the optimizations. The `SAMO.m` code is used in this project with permission from Prof. Viktor Berbyuk from Chalmers University of Technology. `SAMO.m` is the main optimization code that first runs a sensitivity analysis of the defined problem, and then proceeds to run a multi-objective optimization. The purpose of the SAMO code is to generate an optimal set of design variables or design parameters called Pareto set. Those design parameters are associated to the optimal minimal values for the objective functions which are referred to as Pareto front or curve. The SAMO code always solves an optimization problem by only minimizing given objective functions.

The coupling between the fractal energy harvester's FE model in COMSOL Multyphysics and MATLAB is ensured by a COMSOL feature called LiveLink for MATLAB.

Finally, the algorithms processes, used by the optimization procedures for the `SAMO.m` and `MBSD.m` codes for each Phase optimization setup, are detailed as flow charts in Figures 3.5, 3.6, 3.7 and 3.8.

3.1.1 Phase 0 and Phase 1 optimization setup

For the Phase 0 and 1 optimization setup, the first simulation intended to be ran is the static analysis followed by a modal analysis (eigenfrequencies computation). The static load case considers the bottom edge of the fractal tree (see Figure 2.3) to be clamped, whereas the dynamic load case considers the same bottom edge to have a prescribed out of plane displacement. If the maximal static load stress encountered at the bottom of the fractal tree first branch (trunk root) is permissible, the Frequency Domain simulation is carried out to obtain the Frequency Response Functions (FRFs) for each branch's normal stress in the longitudinal direction.

For each optimization setup procedure, the `SAMO.m` MATLAB code updates the design variables for each iteration. Afterwards, the function `MBSD.m` generates the new fractal energy harvester geometry and computes the static analysis, modal analysis and dynamic FRF analysis.

Depending on the optimization Phase, a different version of the function `MBSD.m` MATLAB code is used, and other functions are modified too. There are individual run scripts used for each optimization Phase setup. The script intended to run for the Phase 0 and Phase 1 optimization setup is called `run_SAMO_time_tree_geometry.m`, whereas the script used for the Phase 2 optimization setup is named `run_SAMO_time_Piezo_area.m`.

3.1.2 Phase 2 optimization setup

The Phase 2 optimization setup run MATLAB code `run_SAMO_time_Piezo_area.m` is divided into two sections.

The first section needs an optimized fractal tree geometry shape obtained from the Phase 1 optimization setup to be provided, and if necessary have also the proof masses specifications. The aim of the first section of the script `run_SAMO_time_Piezo_area.m` is to compute the location of maximal von Mises stress per branch. The maximal von Mises stress per branch is computed at a certain reference frequency which is deemed to be optimal according to the von Mises stress FRF for each branch. That reference frequency is located within an optimal bandwidth frequency range. That optimal bandwidth is selected by either favouring a von Mises stress bandwidth with a larger frequency range, or rather a bandwidth with a larger area above stress FRF threshold (see Figure K.1).

The second section of `run_SAMO_time_Piezo_area.m` is identical to the entire code `run_SAMO_time_tree_geometry.m` which runs `SAMO.m`.

3.1.3 FE models and FRF stress threshold with conversion factor

In order to set an appropriate threshold for the total longitudinal stress FRF, a conversion factor of voltage over longitudinal stress for the next coming Phase is computed. The 100 mV voltage threshold is divided by that conversion factor to obtain the stress FRF threshold.

The chosen design from Phase 1 is modeled as a multi-layered FE model in COMSOL Multiphysics (see example in Figure 3.1). An FRF analysis is performed where individual branches surface averaged voltage outputs are extracted and summed together for a total voltage output. This multi-layered FE model considers the presence of piezoelectric material layers as shell elements above the ones representing the fractal tree framework. The voltage for each piezoelectric material section per branch is extracted from the top layer of the multi-layered material sections.

A single layered FE model of the same design with no piezoelectric material consideration is also built in COMSOL (see example in Figure 3.2). From the FRF analysis of that FE model, the individual branches surface averaged longitudinal, or von Mises, stress outputs are extracted and summed together for a total stress output. The stress is extracted from the single layer which is the midplane of the steel framework. For the conversion factor FRF, the longitudinal stress is more appropriate to be used than the von Mises stress.

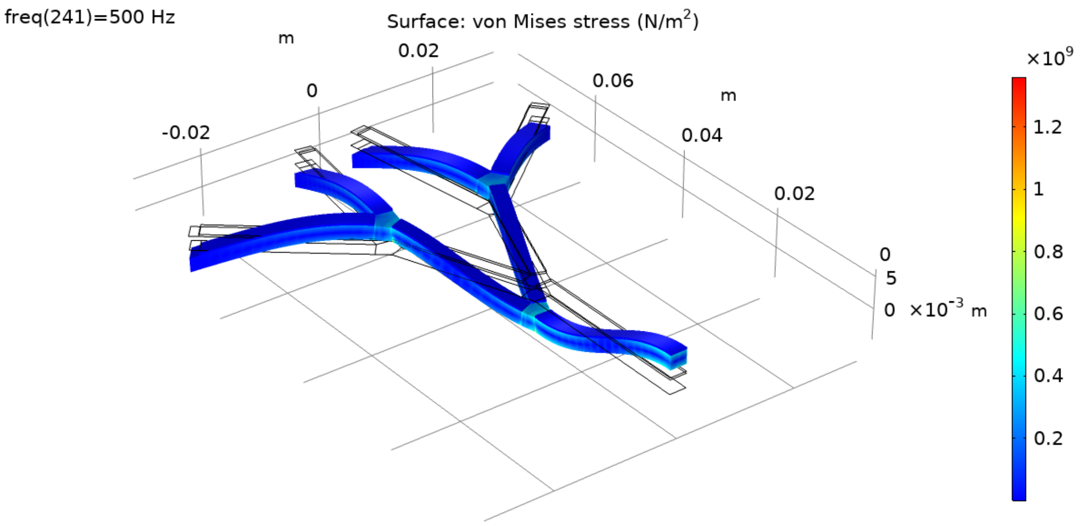


Figure 3.1: Von Mises stress for an excitation frequency of 500 Hz when using an FE model of a typical design without proof masses and with piezoelectric material layers built in COMSOL through the Composite Materials Module.

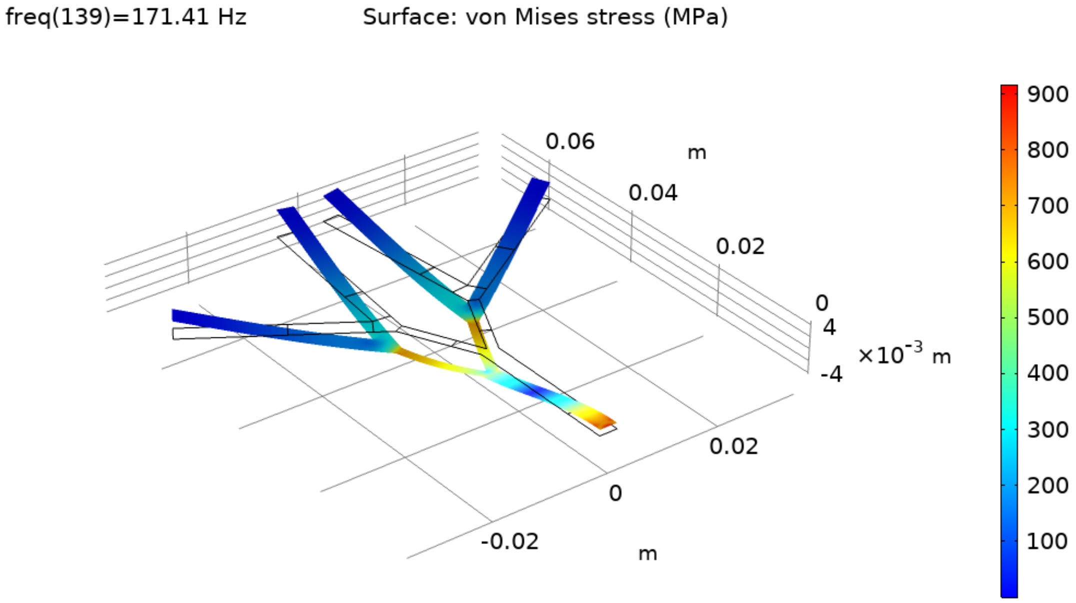


Figure 3.2: Von Mises stress for an excitation frequency of 171.41 Hz when using a single layer FE model of a typical design without proof masses.

For more information about how the FE models are generated in COMSOL and simulation results are extracted, please refer to Appendix L.

3.2 MBSD - optimization problem definition

The name MBSD of this MATLAB function stands for Multi-Body System Dynamics because the developers of the SAMO code intended both the `SAMO.m` and `MBSD.m` MATLAB codes to be used for the optimization of rigid body dynamics systems, for example in terms of dampers and springs selection.

The problem that needs to be defined and optimized is coded by the user as part of `MBSD.m`. For all Phases, `MBSD.m` is the MATLAB function in which the optimization problem is solved for each iteration through performed simulations and computed objective functions.

3.2.1 Input parameters

Before the optimization can be executed, the design variables that are defined in `MBSD.m` have to be controlled, and this is done in a complementary Excel-file named `InputParams.xls`. This file controls the upper and lower bounds of each design variable as well as a few other settings for the sensitivity analysis and the multi-objective optimization.

The `MBSD.m` code takes as input the design variables from `SAMO.m` which are themselves extracted from the `InputParams.xls` Excel file [5]. Note that both the MATLAB code `MBSD.m` and `InputParams.xls` Excel files vary depending on which Phase optimization setup (Phase 0, 1 or 2) simulations are carried out. For additional details regarding the input parameters and GA settings in the Excel-file `InputParams.xls`, please refer to Appendix A.

3.2.2 Fractal energy harvester geometry creation, geometrical conditions and branch sections

For each optimization procedure iteration, the geometry of the fractal energy harvester is modeled in MATLAB as a series of consecutive points which are then imported into COMSOL to form the symmetric tree shape geometry (fractal design). This geometry can have different design parameters (or design variables) that are varied for each iteration during the optimization procedure. Those varied design parameters are the angles between branches, branches lengths, branches widths and additional variables if required, such as proof mass weight and area coverage.

For the Phase 1 optimization setup, the entire shape of the symmetric tree is intended to be optimized by relying on the `SAMO.m` MATLAB code to obtain the Pareto fronts points associated to the objective functions values. For Phase 2, the branch surfaces used for computing surface averaged stress are modified at each optimization iteration. For Phase 1, those branch surfaces are full and do not vary.

3.2.2.1 Fractal tree geometry layout

Initially, a new MATLAB code that generates the geometry of the fractal energy harvester was created. This new code is based on a previous code where a 2D fractal design is generated with certain input parameters that are given. That code had five input parameters known as L, α, w, N and Δ. L is the length of the trunk (first branch). α is the angle between the next branches that extend from their parent branch. w is a constant width that is identical for every branch. N is the numbers of levels for branches extending, where $N = 1$ refers to one branch level extension from the trunk. Δ is a variable which sets the length of the next extending branch pair as $L_{i+1} = L_i/\Delta$.

The new MATLAB code is modified so that a few more individual parameters can be set. However, due to the high complexity of the fractal design geometry, a few of the parameters from the original code are reduced or set to a limit. The first parameter is N, which can now only have a maximum value $N = 2$, i.e. the branches can only extend to maximum two levels. Also, the Δ parameter is now removed. Instead of having the length of the next extending branches depending on their parent branch, new branch length parameters L_2 and L_3 are introduced. L_2 and L_3 are respectively the branch lengths for branch levels $N = 1$ and $N = 2$. Another parameter that is being added is α_3, which defines the angle between the top branches at branch level $N = 2$. That is similar to α which defines the angle between mid branches at branch level $N = 1$. The last parameters added are w_2 and w_3, which designate individual widths of branch levels $N = 1$ and $N = 2$ respectively, and the main trunk branch width is given by w.

The fractal energy harvester is chosen to be symmetric along the axis $x = 0$. This ensures that some modes will be symmetric with respect to axis $x = 0$. If the geometry were to be non symmetric then there would be no mode at all with a symmetric eigenshape. This symmetry can be observed in Figure 1.2.

Finally, Figure 3.3 gives a summary of the overall fractal tree geometry layout, and its design parameters.

3.2.2.2 Piezoelectric material placement

Another geometrical aspect which is important in the design of the fractal tree is the piezoelectric material coverage surface for each branch. As can be seen in Figure 3.3, there is no piezoelectric material coverage at the junctions between branches. It is a good design approach to have separate piezoelectric material coverage sections per branch, because that way the voltage output contribution from each branch will not be interacting with voltage contributions of opposite signs from other branches. Some of the eigenmodes, including the 4^{th} and 5^{th} which are of interest in this project (see Figure 5.12 and 5.13 in the *Results* section), are not purely bending modes, and this implies that they are also twisting modes. This means that when one branch connected to the same junction bends upwards (compression on upper surface), the other one bends downwards (tension on upper surface). The junction of those two branches is solicited in both compression and tension and it is not an area to be covered by piezoelectric material. This ensures that there is no piezoelectric material to cover the fractal tree areas where there are "S" shape deformations. Simply put, such undesired "S" shape deformations are associated to both compression and tension surface stresses and they result in undesired voltage outputs of opposite sign.

To ensure that no voltage cancelling effect, or the least of it, occurs for the Phase 2 optimization setup, the GA tries to find a final design solution which avoids having the occurrence of "S" shape deformations within a piezoelectric material branch surface. Namely, for Phase 2, the piezoelectric material placement is optimized by considering an area coverage variation for each branch of the fractal tree design. This is carried out in order to favour the least negative voltage output generation per branch and a maximal longitudinal stress in the Frequency Domain up to 200 Hz.

3.2.2.3 Proof masses placement

Since adding proof masses to the fractal energy harvester can change the behaviour and the resulting FRF output, another parameter is also added to control this in the fractal tree geometry. In this project it is called "Proof mass ratio" and it is a proportion value dependent on the length L_3 of the top branches. This will set an equal area on each top branch's end where the proof masses are intended to be placed. Having the "Proof mass ratio" set to zero, means having no proof mass at all. Lastly, the actual mass of the proof masses is controlled in the COMSOL FE model through the *Added mass* feature.

3.2.2.4 Fractal tree geometry and associated MATLAB codes

The 2D geometry of the fractal energy harvester is generated by running the MATLAB function called `Tree_code_function.m` for Phase 0 and Phase 1, and `Tree_code_function_modified.m` for Phase 2. Figure 3.3 shows the overall shape and dimensions of the fractal energy harvester that is being generated. In other words, for the Phase 1 optimization setup procedure, the MATLAB function `Tree_code_function.m` is responsible for generating the overall tree geometry and also the branch surfaces. These are the surfaces (shown in orange in Figure 3.3) of each individual branch where the piezoelectric material is intended to be placed on. They are also the surfaces where the mechanical stress and voltage data will be extracted from.

For the Phase 2 optimization setup, the modified version of the geometry code, `Tree_code_function_modified.m`, uses additionally as function inputs proportions of piezoelectric material area coverage for each branch (shown in orange in Figure 3.4). This results in different individual branch areas at each Phase 2 optimization procedure iteration. Unlike the case for the Phases 0 and 1 optimization setup, the fractal tree geometry remains unchanged for the Phase 2. There is only a variation of parameters associated to the proportions of branch surfaces which are covered by the piezoelectric material, refer to Figure 3.4.

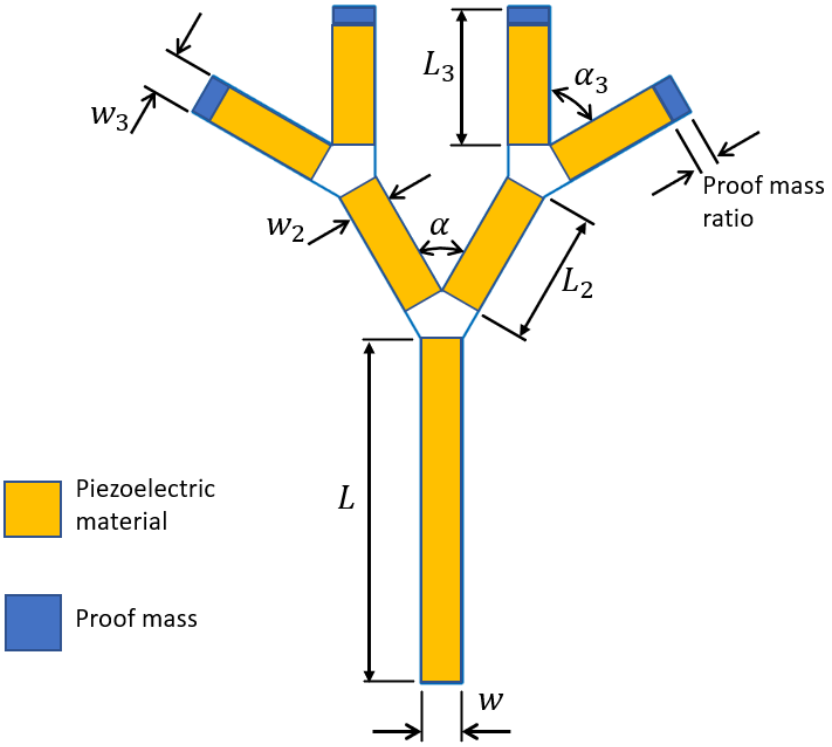


Figure 3.3: The symmetric 2D geometry of the fractal energy harvester with two levels of branches, and its input parameters for Phase 0 and 1 optimization setups.

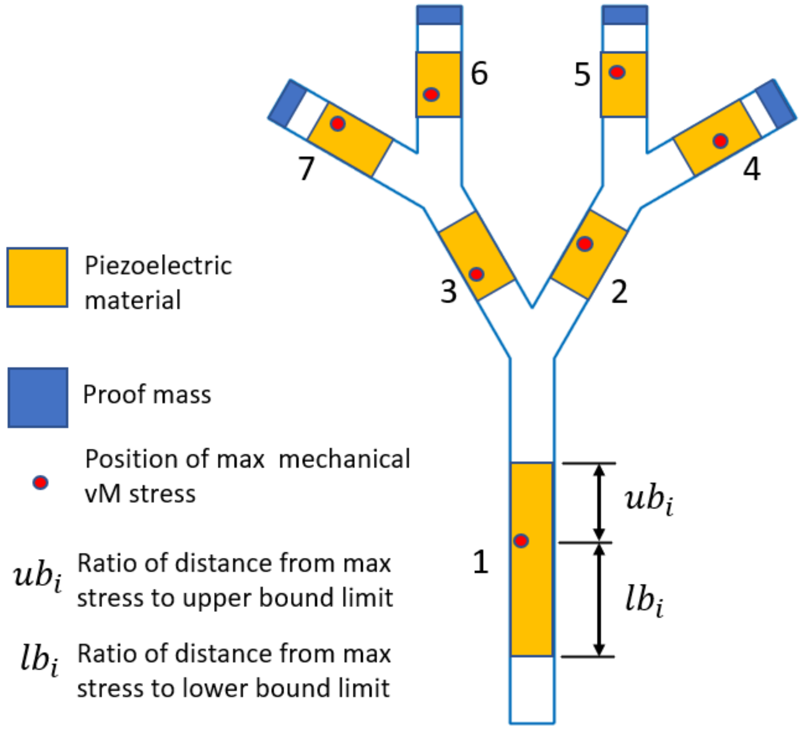

Piezoelectric material
Proof mass
Position of max mechanical vM stress
ub_i Ratio of distance from max stress to upper bound limit
lb_i Ratio of distance from max stress to lower bound limit
1
2
3
4
5
6
7
ub_i
lb_i

Figure 3.4: The symmetric 2D geometry of the fractal energy harvester with two levels of branches, and its input parameters for Phase 2 optimization setup. Index i refers to the branch number from 1 to 7.

3.2.2.5 Corners determined by intersecting lines

Additionally, within the MATLAB function `Tree_code_function.m` a function named `fintersect.m` is responsible for determining an intersection point between two straight curves each defined by equation $y = ax + b$. Likewise, within the MATLAB function `Tree_code_function_modified.m`, a modified version of `fintersect.m` is also used, `fintersect_modified.m`. Useful descriptions about the equations used to compute intersection points can be found in Appendix C.

3.2.2.6 Branch numbering

Regardless of the Phase optimization setup, the numbering of branches 1 to 7 is done from right to left branches and starting from the bottom branch (trunk). For instance, the top left branch is the 7^{th} branch and the top right one is the 4^{th}. This numbering of branches is visible in Figure 3.4.

3.2.2.7 Geometrical conditions consideration for Phases 0 and 1

During the Phase 0 and Phase 1 optimization setups procedures, there is a risk that the fractal tree geometry that is being generated at any iteration by the GA might extend outside certain geometric limiting bounds. There is also a risk that a geometry generated by the GA is featuring nonphysical dimensions, e.g. the width of the first branch being wider than the second level branches. In order to minimize the risk of having an error or collecting undesired results during an optimization procedure, a fail-safe feature is implemented in the geometry generation code in form of geometrical conditions. A value of 1 is assigned to a geometrical condition if a geometry fails a specific requirement and 0 if it does not. Undesirable angles, a geometry with top branches outside the 8 cm × 8 cm design envelope, the crossing of middle top branches, or branches being too wide, are some examples of the geometrical conditions that are being checked. If the geometry fails any of these requirements, another output variable called objective function condition `OFC` (see Appendix E) takes into account all of the geometrical conditions and indicates in the optimization procedure that a specific iteration must be skipped for any further COMSOL analysis. Appendix D shows a few examples of unwanted geometries that this fail-safe feature would capture.

3.2.2.8 Piezoelectric material placement parameters for Phase 2

Particularly with respect to the Phase 2 optimization setup, for each branch, there is a distance from the location of maximal von Mises stress (at a certain reference frequency) to the upper limit of the total branch surface (see Figure 3.4). This distance can be multiplied by an upper bound proportion, referred to as ub. Equivalently, there is also a distance from the location of maximal stress to the bottom limit of the total branch surface. This distance can be multiplied by a lower bound proportion, referred to as lb. This notation is used to identify for each branch of index i, the upper and lower bounds proportions of piezoelectric material coverage from the location of the maximal stress. Finally, the distance from the point of maximal stress to the total branch surface upper bound is noted ub_i and to the lower bound lb_i.

3.2.3 Static analysis for Phases 0 and 1

For the Phase 0 and Phase 1 optimization setup procedure, the first case study that is being checked is the static load of 10 g-force. This is to determine if the fractal energy harvester can withstand a sudden impact such as being accidentally dropped to the ground. For each optimization procedure iteration, the static analysis, that is set up in COMSOL (see section about COMSOL in Appendix L), is performed in `MBSD.m` via a MATLAB function called `run_comsol_static_function.m`. For the Phase 2 optimization setup there is no need for a static analysis to be carried out because the two optimal general tree designs, with and without proof masses, have already been selected. Appendix G shows more information about the static analysis settings used for the Phase 0 and 1 to be able to export simulations results from COMSOL and import them in MATLAB.

For the Phase 0 and Phase 1, if the stress in the structure exceeds the stress limit of the material (yield stress) divided by a security factor (SF), the optimization procedure will discard the fractal energy harvester design and skip the next COMSOL analysis. In that case, the objective functions are given extremely high values (10^{10}) which will signal to SAMO that the combined input parameters are undesirable. However, if the stress is under the limit of yield stress divided by the security factor, the design is approved and will be used in the eigenfrequency (modal analysis) and FRF studies. The value of the objective function condition `OFC` variable also indicates whether or not a design will possibly break due to the static load, see Appendix E for clarifications.

Also, the maximum absolute value of the displacements at the free end corners of the top branches are computed in COMSOL, and then exported and displayed in the MATLAB command window to see if they are within a reasonable range. They are not checked for any specific limit, but this can be added as a parameter in future studies.

3.2.4 Eigenfrequency study

As mentioned previously, for the Phase 0 optimization setup only two eigenfrequencies intended to be minimized are set as objective functions. To do so, the first 10 eigenfrequncies are loaded in MATLAB via the function `Loading_Comsol_mat_files_MBSD_eig_study_only.m`. The eigenfrequency study is also mandatory so that the FRF analysis can be performed by the Phase 1 and Phase 2 optimization setups. Appendix H provides more information about how the eigenfrequencies for all the Phases are exported from COMSOL and imported in MATLAB.

3.2.5 FRF analysis for Phases 1 and 2

For Phase 1 and Phase 2 optimization setups, after the modal analysis (eigenfrequency study) is performed, the next step of the optimization procedure is an FRF study computed in COMSOL. For each optimization procedure iteration, the Frequency Domain (or FRF) analysis is performed in `MBSD.m` via the MATLAB function `run_comsol_eig_frf_study_function.m` for the Phase 1, and `run_comsol_eig_frf_study_function_PIEZO.m` for the Phase 2. Before proceeding officially with the Phase 2 optimization setup, which is ran in the second section of the main code `run_SAMO_time_Piezo_area.m`, the first section of that code must determine the location of maximal von Mises stress for each branch at a reference frequency. To do so, the function `MBSD_modified.m` calls the function `run_comsol_eig_frf_study_function_modified.m`. Appendix J specifies FRF analysis settings used for both Phase 1 and Phase 2 to export simulations results from COMSOL and import them in MATLAB.

3.2.5.1 Modal analysis prior to FRF analysis

The modal analysis is essential to be performed so that two chosen eigenfrequencies f_1 and f_2, here 4^{th} and 5^{th} eigenfrequencies, can be used to characterize the Rayleigh damping. The modal analysis prior to the FRF study is also necessary for a geometric progression function to be able to generate concentrated frequency samples around eigenfrequencies.

3.2.5.2 Frequency samples generated by a geometric progression function

Regarding the Phase 1 optimization setup, the range of frequencies to be studied in the FRF anlsysis was initially set to be from 5 Hz to 200 Hz with a step of 1 Hz. However, since large FRF outputs mostly occur close to the eigenfrequencies, the initial frequency step of 1 Hz was considered too coarse. When the frequency step is too large, it can provide not only less accurate FRF outputs, but also a less accurate approximation of FRFs characteristics. The problem is that if the frequency step would be decreased, for example to 0.5 Hz, the FRF analysis would require more computational time.

In order to increase the accuracy of the results for both Phases 1 and 2, without adding unwanted computational time, a MATLAB function which defines the frequency steps is created. This function is called `geometric_prog_function.m` and it generates unevenly spaced frequency samples which are saved in a .txt file. The steps between the frequency samples for the FRF analysis are coarser when further away from an eigenfrequency and finer when approaching an eigenfrequency where there is a large FRF gradient. Appendix I describes the idea of how this MATLAB function works by relying on a geometric progression function.

3.2.6 FRF characteristics as objective functions to be maximised for Phases 1 and 2

Furthermore, there are multiple FRFs characteristics which can be selected as objective functions, and those vary for the Phase 1 and Phase 2 optimization setups. Some FRFs characteristics are also useful prior to carrying out simulations of the Phase 2 optimization setup. All of those different FRFs characteristics are elaborated in the following sections.

On a side note, Appendix K describes how FRF characteristics are computed in MATLAB and what values are objective functions attributed when an FRF does not generate a cumulative longitudinal stress response above a given threshold, i.e. no bandwidths.

3.2.6.1 MATLAB codes used for computing FRFs characterstics

After being imported in MATLAB, the FRF .mat results can be loaded in MATLAB's workspace via the MATLAB function
`Loading_Comsol_mat_files_MBSD_FRF_study_only.m` for Phase 1 and
`Loading_Comsol_mat_files_MBSD_FRF_study_only_PIEZO.m` for Phase 2 which are called in `MBSD.m`. Thus the FRF characteristics, such as the sum of bandwidths frequency ranges and the area over a given threshold, can be maximized and set as objective functions.

The FRF characteristics which are intended to be maximized must be transformed into objective functions which the SAMO code will minimize instead. Those FRF related objective functions which are intended to be minimized are the inverse of the ones intended to be maximized ($OF_{min} = 1/OF_{max}$).

The threshold is a pressure value in Pa which is obtained via a conversion factor applied to the the 100 mV voltage threshold. Consequently, for the Phase 1 optimization setup, within `Loading_Comsol_mat_files_MBSD_FRF_study_only.m`, the MATLAB function `Find_bandwidth_Matrix_bandwidths_areas_STRESS.m` provides FRF characteristics as outputs. Similarly, for the Phase 2 optimization setup, the same function
`Find_bandwidth_Matrix_bandwidths_areas_STRESS.m` provides FRF characteristics within the function `Loading_Comsol_mat_files_MBSD_FRF_study_only_PIEZO.m`.

3.2.6.2 Localization of maximal von Mises stress per branch section prior to simulations for Phase 2

Exceptionally, prior to starting the Phase 2 optimization setup simulations, there is another MATLAB function which is used for determining stress surface averaged FRF characteristics. That function is called
`Find_bandwidth_Matrix_bandwidths_areas_modified.m` and it is used for determining the reference frequency and the location of maximal von Mises per branch surface at

that specific frequency. The function is used in the `Loading_Comsol_mat_files_MBSD_FRF_study_only_modified.m` MATLAB script which itself is called by the `MBSD_modified.m` MATLAB function. The MATLAB function `Find_bandwidth_Matrix_bandwidths_areas_modified.m` determines that reference frequency within an optimal bandwidth frequency range.

For the Phase 2 optimization setup, an optimal bandwidth from each branch's von Mises stress FRF must be selected. The optimal bandwidth can be selected either by favouring the largest bandwidth or rather a bandwidth with the largest area above stress FRF threshold. Once the optimal bandwidth is selected, a reference frequency remains to be picked within that frequency range. Now, the reference frequency selection is different whether there is a single eigenfrequency with one FRF peak, or if there are multiple eigenfrequencies with corresponding FRF peaks. If there is a single eigenfrequency with one FRF peak within the optimal bandwidth, then it is obvious that this eigenfrequency is a good reference. However, when there are more eigenfrequencies in that bandwidth frequency range, it is better to consider the FRF dips in between peak values, which are FRF local minima. If there is only one FRF local minimum value located between two FRF peaks, the reference frequency is where that FRF local minimum occurs. If there happens to be more local minima FRF values with multiple FRF dips occurring within the bandwidth frequency range, then the frequency at which the highest local minimum FRF value occurs is selected as the reference frequency. In other words, the frequency associated to the maximal FRF value amongst local minima is selected as the reference frequency.

3.2.6.3 Conversion factor used for voltage threshold conversion

Finally, the cumulative stress FRF threshold considers each branch's surface averaged longitudinal stress FRFs values. It is obtained via a conversion factor of voltage over stress which is computed in `MBSD_modified.m` by `conversion_factor_stress2volt_function.m`. More details can be found in Appendix L section L.7 about how this scalar conversion factor is computed .

3.2.7 Flow charts for `SAMO.m` and for Phases 0, 1 and 2

The following Figures 3.5, 3.6, 3.7 and 3.8 present flow charts of the MATLAB script `SAMO.m`, and of different `MBSD.m` variants for all Phases optimization setups.

Figure 3.5: Flow chart of `SAMO.m`

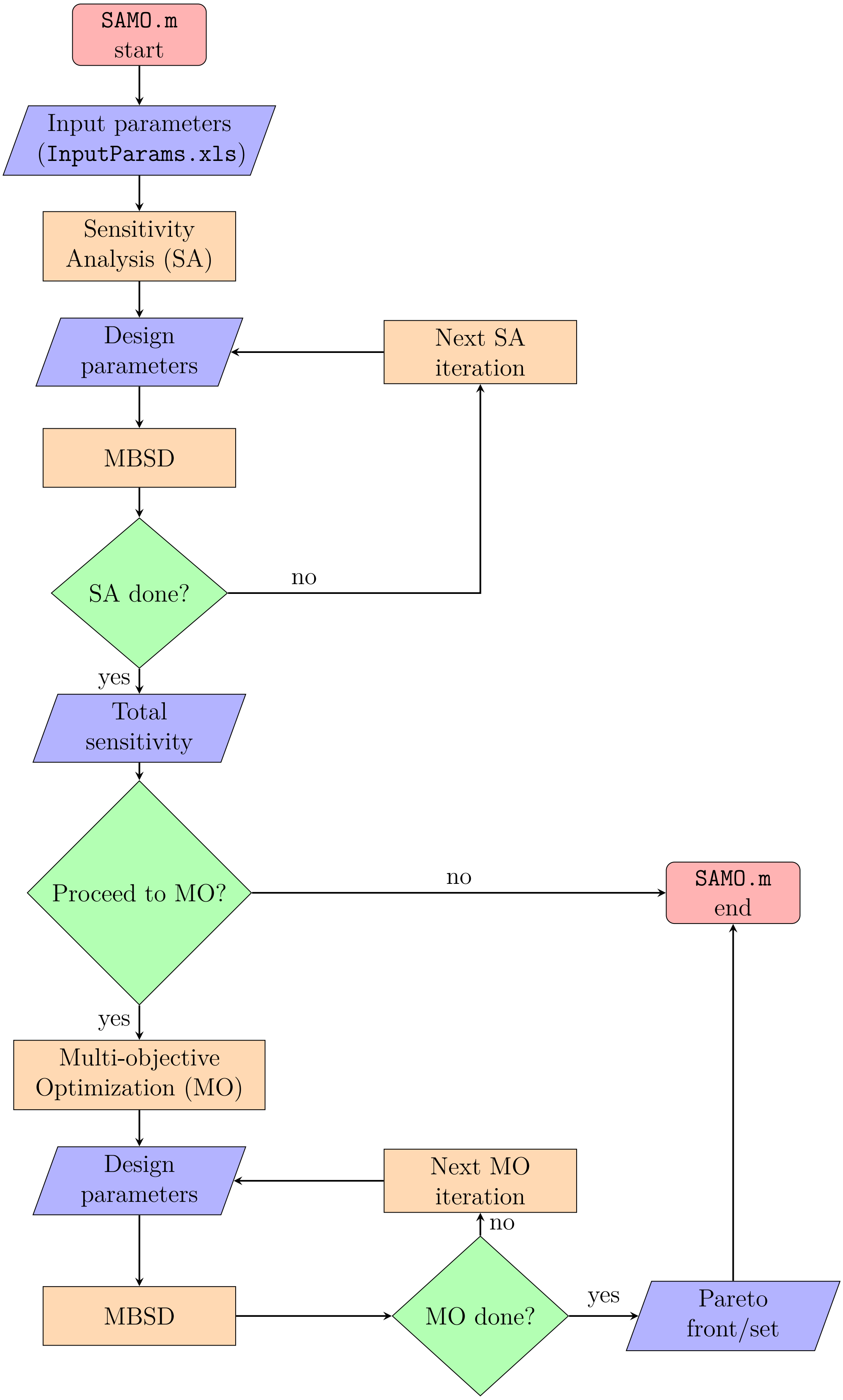


Figure 3.6: Flow chart of `MBSD.m` for Phases 0 and 1 of the fractal tree design optimization setup. Phase 0 has no FRF analysis, only Phase 1 does. The run code which calls this `MBSD.m` script is `run_SAMO_time_tree_geometry.m`.

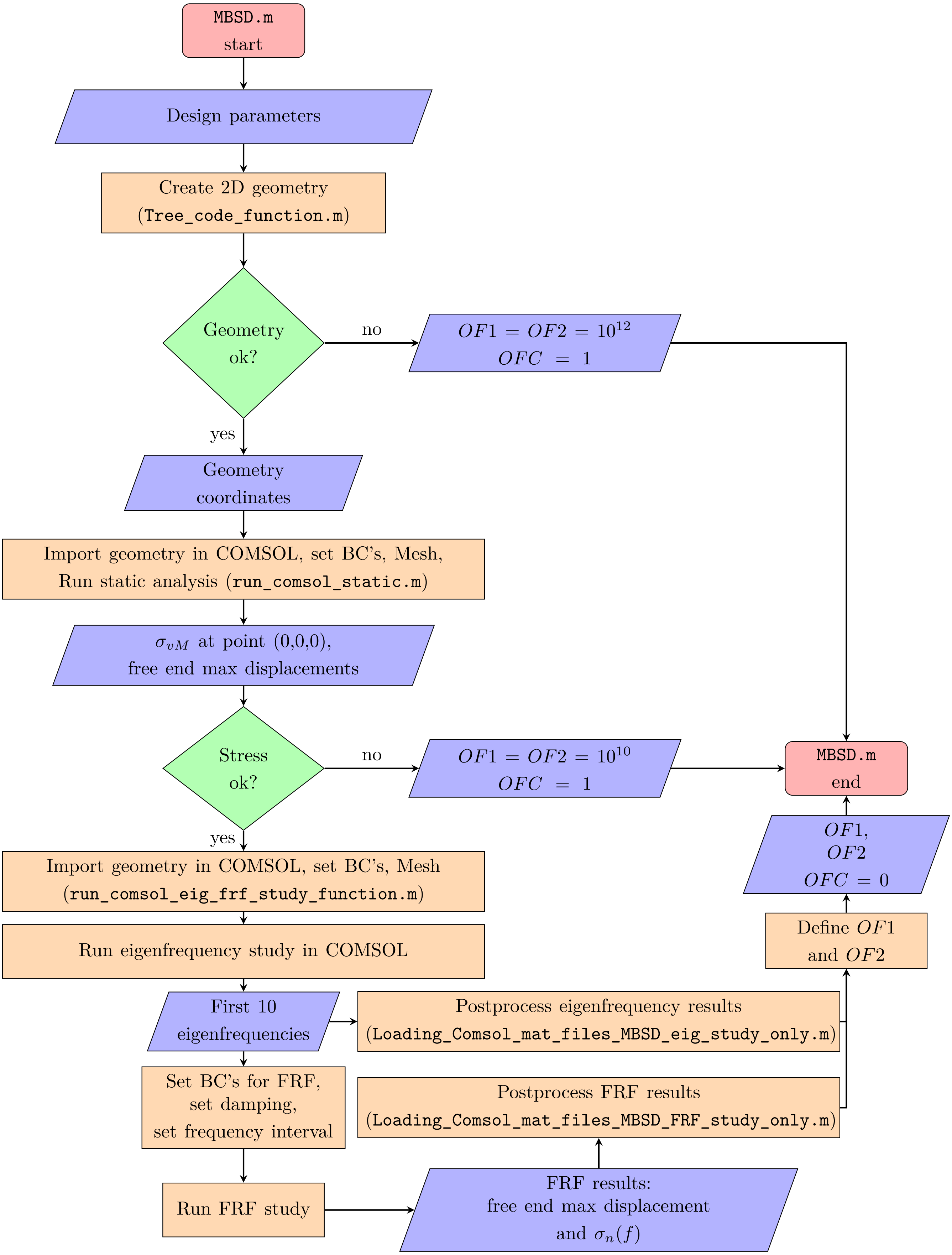

MBSD.m
start
Design parameters
Create 2D geometry
(Tree_code_function.m)
Geometry ok?
no
$OF1 = OF2 = 10^{12}$
$OFC = 1$
yes
Geometry coordinates
Import geometry in COMSOL, set BC's, Mesh,
Run static analysis (run_comsol_static.m)
σ_{vM} at point (0,0,0),
free end max displacements
Stress ok?
no
$OF1 = OF2 = 10^{10}$
$OFC = 1$
MBSD.m
end
yes
Import geometry in COMSOL, set BC's, Mesh
(run_comsol_eig_frf_study_function.m)
$OF1$, $OF2$
$OFC = 0$
Run eigenfrequency study in COMSOL
Define $OF1$ and $OF2$
First 10 eigenfrequencies
Postprocess eigenfrequency results
(Loading_Comsol_mat_files_MBSD_eig_study_only.m)
Set BC's for FRF,
set damping,
set frequency interval
Postprocess FRF results
(Loading_Comsol_mat_files_MBSD_FRF_study_only.m)
Run FRF study
FRF results:
free end max displacement
and $\sigma_n(f)$

Figure 3.7: Flow chart of `MBSD_modified.m` prior to Phase 2 of the fractal tree design optimization setup. The run code which calls `MBSD_modified.m` script in its first section is `run_SAMO_time_Piezo_area.m`.

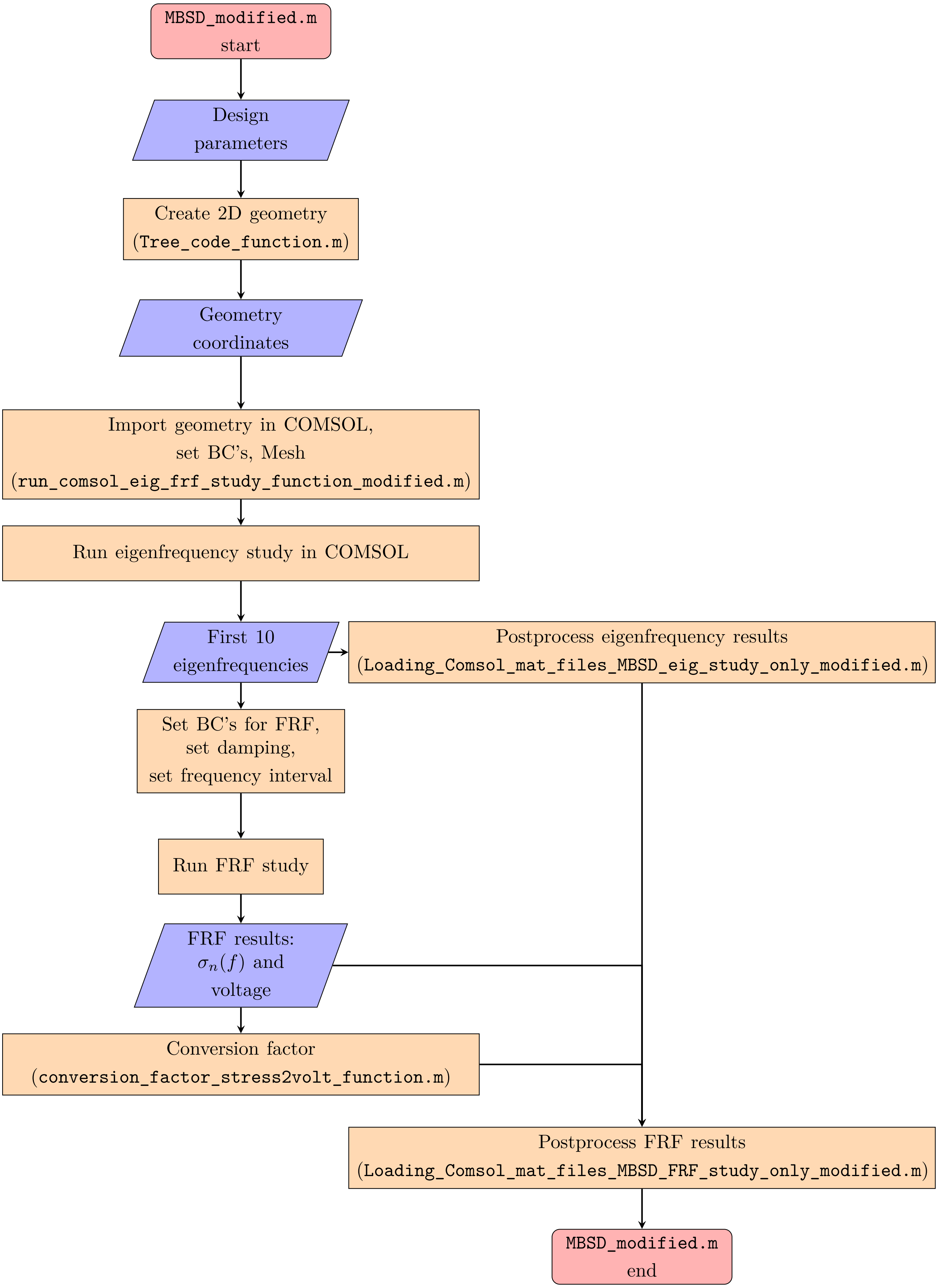
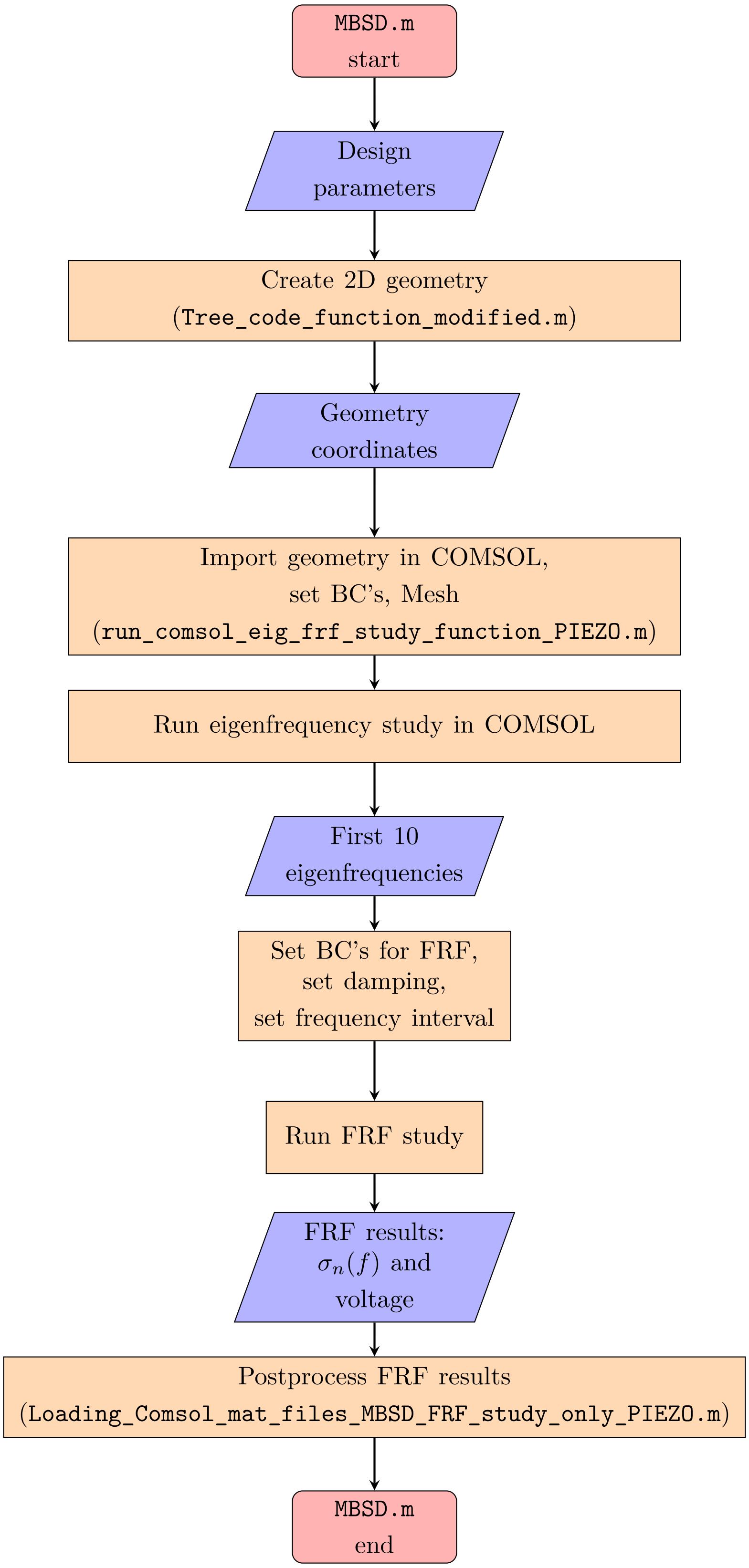

MBSD_modified.m start
Design parameters
Create 2D geometry (Tree_code_function.m)
Geometry coordinates
Import geometry in COMSOL, set BC's, Mesh (run_comsol_eig_frf_study_function_modified.m)
Run eigenfrequency study in COMSOL
First 10 eigenfrequencies
Postprocess eigenfrequency results (Loading_Comsol_mat_files_MBSD_eig_study_only_modified.m)
Set BC's for FRF, set damping, set frequency interval
Run FRF study
FRF results: $\sigma_n(f)$ and voltage
Conversion factor (conversion_factor_stress2volt_function.m)
Postprocess FRF results (Loading_Comsol_mat_files_MBSD_FRF_study_only_modified.m)
MBSD_modified.m end

Figure 3.8: Flow chart of `MBSD.m` for Phase 2 of the fractal tree design optimization setup. The run code which calls `MBSD.m` script in its second section is `run_SAMO_time_Piezo_area.m`.



4

Assumptions

The following chapter presents the numerical and modelling assumptions which are considered for the optimization setups.

4.1 Fractal tree geometry structure is thin

The main assumption which is considered for all simulations is that the steel structure which is the framework of the fractal energy harvester is very thin in comparison to its thickness and branch lengths. The same can be said about the PVDF piezoelectric material layers, however those have a mass and stiffness which are neglected in the optimization FE model used for the static analysis and for the longitudinal stress and displacements FRF computation. Consequently, thin structures can be modeled as shell elements rather than solid elements which have a considerable thickness. In the FE model used for voltage FRF computation, the piezoelectric material layers do have indeed an added mass and stiffness, and they are modeled as multi-layered shell elements with an offset above the shell elements which constitute the steel framework of the fractal energy harvester. That being said, for the FE model used for stress FRF analysis, only the steel framework is modeled as shell elements and that saves a considerable amount of computational time for simulations at each optimization iteration. Not taking into account the voltage FRF for both Phases, but instead only the longitudinal stress FRF saves considerably an additional computational time for simulations at each optimization iteration.

This assumption has most notably an impact on eigenfrequencies, but also overall on FRFs values. For all Phases optimization setups, the eigenfrequencies are imported first in MATLAB's workspace before carrying out the FRF analysis. For the Phase 2, the eigenfrequencies are also computed at each optimization iteration even though they remain constant. However, if the added mass and stiffness of the piezoelectric material were to be considered in the COMSOL FE models used for the Phases 1 and 2 eigenfrequency study, then the eigenfrequencies would vary for each optimization iteration. This would occur, because at each optimization iteration the different piezoelectric material area coverage for each branch would modify both the mass and stiffness of the FE models, which would in turn affect the eigenfrequencies results. Yet, the current FE models used for both Phases 1 and 2 neglect the mass and stiffness of the piezoelectric material layers since they are very thin and made also of a light material, PVDF layers of 0.028 mm.

4.2 Neglected added momentum

When considering a design with added proof masses at the end of the top branches, only the additional mass is considered in the FE models while added stiffness is disregarded. An important observation is that when proof masses and piezoelectric material layers are not modelled at all, not even as shell elements, the additional momentum is neglected due to their negligible weight. Otherwise, the center of gravity for the layers of piezoelectric materials and proof masses would be located at the midplane of their thickness, and would contribute to the total momentum.

4.3 Mesh is refined and there is no need for mesh convergence

For all FE models, the utilized mesh for each optimization iteration is assumed to be refined enough and equivalent to the mesh which would have been obtained if a mesh convergence were to be carried out for each optimization iteration.

4.4 Surface averaged data exported directly from COMSOL is almost equivalent to being computed in MATLAB

Another computational assumption is considered for each optimization iteration when extracting from COMSOL the surface averaged stress FRF data. For all Phases, the mesh seemed to be quite regular and having nodes being almost equidistant in the zones of interest. The zones of interest are the surfaces from which stress data is exported for each node and then averaged through the MATLAB function *mean*. This entails that the nodal stress values for σ_x, τ_{xy} and σ_y, which are used to compute the surface averaged longitudinal stress σ_n, have all the same weight. This is however not the case when post-processing results of the final design configurations for both Phase 1 and Phase 2. When post-processing results, it is desired to have the best attainable accuracy since saving computational time is not relevant. Consequently, the COMSOL feature *Average* is used to compute more accurately the surface averaged values directly in COMSOL with nodal weight consideration.

4.5 Simplified damping

Regarding FRFs results, both in terms of stress and voltage, they are highly dependant on damping, particularly on the two chosen modes used for Rayleigh damping characterization. For typical fractal tree geometries, the 4^{th} and 5^{th} modes have shown to produce desired stress FRF characteristics. One of those characterstic is to have both eigenfrequencies below the 200 Hz limit of analysis. Another desired FRF characteristic of the 4^{th} and 5^{th} modes is that they are bending modes with reduced twisting mode shapes (see Eigenmodes Figures 5.12 and 5.13 in the *Results* section). The 4^{th} and 5^{th} modes were first analysed for a common or typical fractal tree geometry, see Appendix M. Although they look similar to the modes obtained in the *Results* section, they are different in shape due to a different geometry. Bending modes produce higher stress FRF values and thus higher voltage output making them desirable modes to extract energy from. For this reason, these two bending modes, which generate the highest stress FRF values and are relatively close to each other in terms of eigenfrequencies, are used for Rayleigh damping characterisation. Nevertheless, damping characterisation can be furthermore improved by considering added damping (with loss factor) where piezoelectric material and proof masses would be deposited. This additional damping would be due to the adhesive used to join the piezoelectric material layers and the proof masses to the steel framework structure of the fractal tree design.

4.6 Lateral motion contact is not verified

The static (10 g-force) and dynamic (2 g-force) loads for each optimization iteration in Phase 0 and 1, were only applied in the out of plane direction. These load cases do not check lateral motion of the tree for any other load applied laterally (in-plane). If the top branches are too close to each other, there is a possibility that their corners could touch each other for a lateral static or dynamic load case. One of the two designs, either with or without proof masses, would be better than the other one according to this criteria.

5

Results and discussion

In this chapter the results of all optimization Phase setups (Phase 0, 1 and 2) are presented.

5.1 Mechanical properties and settings

The fractal tree geometry frame is set to structural steel in COMSOL by being selected from COMSOL's material library, and its yield strength is 240 MPa. The steel plate thickness is constant with a value of 0.2 mm, which is the thickness of the plates provided by the manufacturer. Regarding damping, which is characterized by the Rayleigh damping model, two eigenfrequencies f_{d1} and f_{d2} can be chosen. The damping ratios associated to the corresponding modes are set to $\zeta_{d1} = \zeta_{d2} = 0.001$. The magnitude of Rayleigh damping was not investigated, but rather only the influence of it being applied to two specific modes, namely to 4^{th} and 5^{th} eigenmodes.

5.2 Mesh

Concerning the fractal tree energy harvester mesh, there was no mesh convergence carried out for neither static, modal or FRF dynamic analysis. There was no mesh convergence performed for a typical single structural case of fractal tree geometry because the Rayleigh damping is approximate and the geometry changes at each optimization iteration for Phases 0 and 1. Ideally, if there were to be a mesh convergence carried out, it would take into account a voltage FRF instead of the stress FRF which is currently used for optimization. The mesh convergence was ignored, also because it would require too much computational time to do it for every optimization iteration for which the fractal tree geometry is changed during Phases 0 and 1.

5.2.1 Mesh control

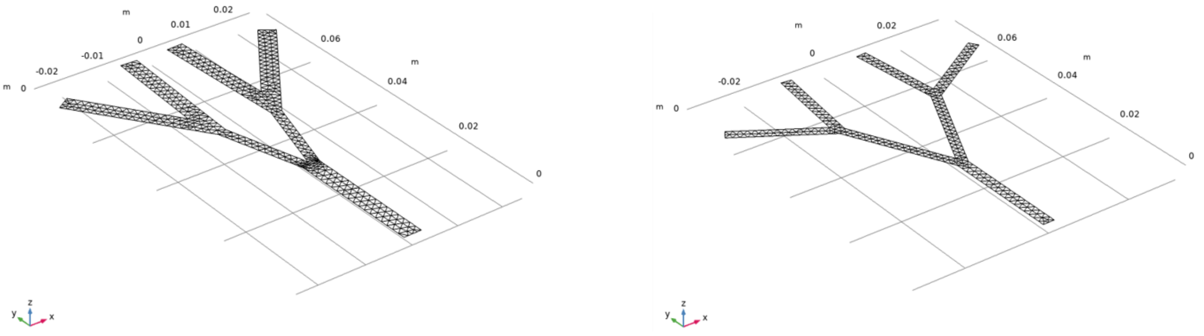
As an alternative, the mesh was set with COMSOL's default *Physics-controlled mesh* setting with element size as *Extremely fine*. Even though this setting suggests that the mesh would be extremely fine, it was never ensured if this default setting is accurate enough for all geometrical cases, and if a manually adjusted finer mesh would be needed. Nevertheless, this mesh setting only increased computational time per optimization iteration by a few seconds compared to using *Normal* element size during initial test runs of the optimization MATLAB code scripts. Thus, the *Extremely fine* mesh element size was used for all optimization runs and for post-processing optimization results.

5.2.2 Mesh convergence option

Mesh convergence parameters are included in all of the `MBSD.m` code variants for the Phases optimization setups (including in `MBSD_modified.m`), as well as in the simulations running functions, which are **`run_comsol_eig_frf_study_function.m`**, **`run_comsol_eig_frf_study_function_modified.m`**, and **`run_comsol_eig_frf_study_function_PIEZO.m`**. Extensive mesh convergence studies can be included in future optimization runs if needed.

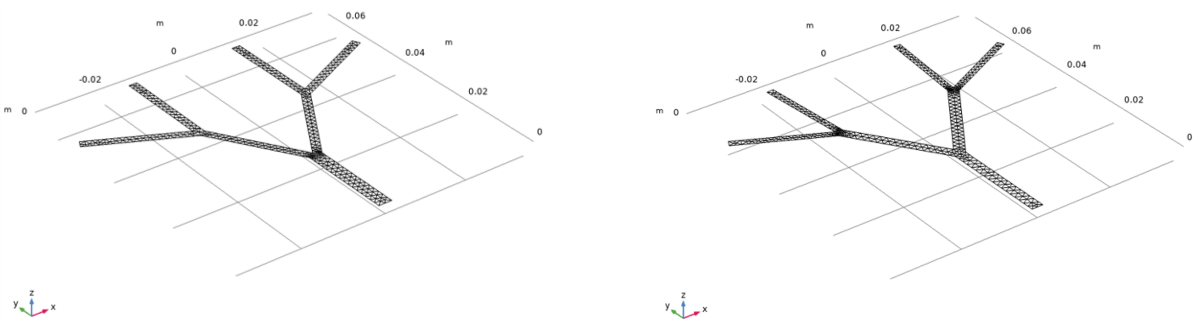
5.2.3 Meshing examples with 2D shell elements

In Figures 5.1 and 5.2 are presented some examples of meshes at some arbitrarily chosen iterations for different designs with proof masses and without proof masses that are being tested by the GA during the Phase 1 optimization setup procedure. As mentioned earlier in the *Assumptions* section, the fractal energy harvester can be considered as thin, meaning that the thickness is considerably smaller than the other dimensions of the geometry. Therefore, the FE COMSOL models used for stress and voltage FRFs consist of 2D shell elements. This reduces the computational cost when solving the FE models while still producing good approximations of solutions.



(a) **(b)**

Figure 5.1: Two mesh examples consisting of 2D Shell elements for two different fractal tree geometries without proof masses tested by a `SAMO` optimization procedure iteration.



(a) **(b)**

Figure 5.2: Two mesh examples consisting of 2D Shell elements for two different fractal tree geometries with proof masses tested by a `SAMO` optimization procedure iteration.

5.3 Phase 0 optimization setup - Minimization of 1^{st} and 5^{th} eigenfrequencies

In an early stage of the project, initial test simulations of eigenfrequency analysis were performed with COMSOL. The design of the fractal energy harvester was set with common design parameters. Based on the simulations, the shapes of the eigenmodes were analyzed, and it was assumed that other fractal energy harvester designs would have similar shapes and behaviours at their corresponding eigenfrequencies.

5.3.1 Objective functions

The 1^{st} and 5^{th} eigenfrequencies are associated to energy harvester modes which are bending and twisting and strain the piezoelectric material to produce a considerable voltage output. It was concluded that the 1^{st} and 5^{th} modes would be the best candidates for the Phase 0 optimization setup. These eigenfrequencies were set to be the objective functions to be minimized. The 1^{st} and 5^{th} eigenfrequencies are usually within the 0 to 200 Hz frequency range. In Appendix M some examples of the first 5 eigenmode shapes are presented.

5.3.2 Input parameters

In the Phase 0, all eight geometry parameters, portrayed in Figure 3.3, were set to be the optimized design parameters. Table 5.1 shows the nominal values, the upper and lower bounds, as well as other input settings for each design parameter. For the Phase 0 optimization setup, the fractal tree design which was analyzed has no added proof masses. Just like for the following Phases 1 and 2, the Rayleigh damping was characterized by considering the 4^{th} and 5^{th} eigenfrequencies with damping ratios of 0.001 for both modes.

Table 5.1: Input parameters for Phase 0 optimization setup.

	L [mm]	L_2 [mm]	L_3 [mm]	α [°]	α_3 [°]	w [mm]	w_2 [mm]	w_3 [mm]
Nominal	20.94	20.94	20.94	60	60	2.5	2.5	2.5
COV	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05
No. Pts	5	5	5	5	5	5	5	5
Dist	N	N	N	N	N	N	N	N
Lower Bnd	10	10	10	20	20	1.5	1.5	1.5
Upper Bnd	30	30	30	80	80	4.0	4.0	4.0
Population Size	25							
No. of generations	25							
Elite Count	4							
Pareto Fraction	1							

5.3.3 Pareto front

The results of the optimization for Phase 0 are shown in Figure 5.3, where a Pareto front is plotted. As can be seen, the 1^{st} eigenfrequency is minimized close to 15 Hz in all cases, while the 5th eigenfrequency varies between 150 Hz and 165 Hz. Full details about the Pareto front and its corresponding Pareto set can be found in Appendix N.

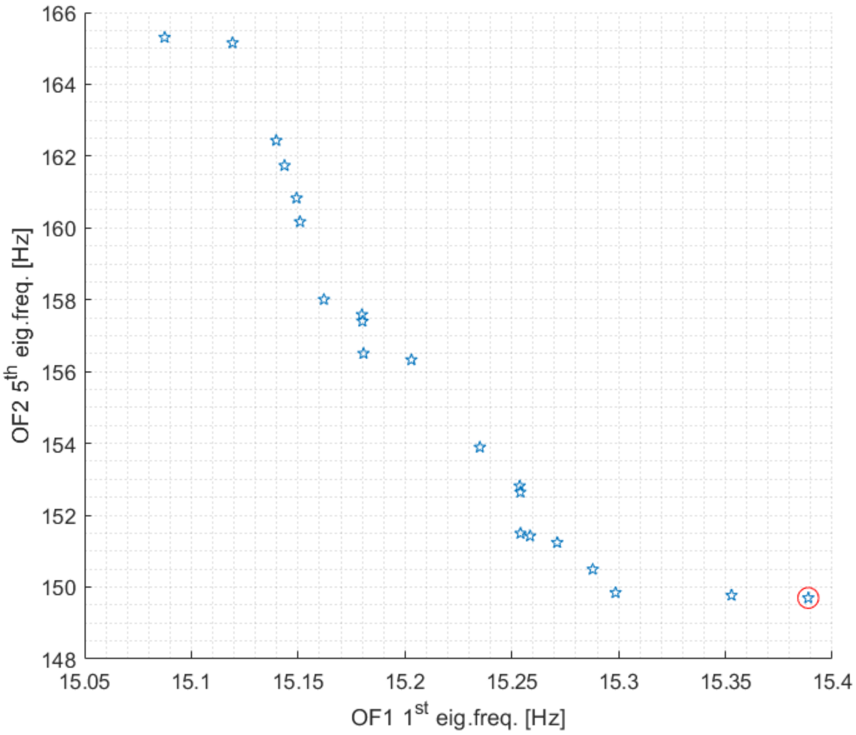


Figure 5.3: Phase 0 Pareto front from optimization of 1st eigenfrequency minimization (OF1) and 5th eigenfrequency minimization (OF2). Red circle shows the chosen design with parameters used as nominal values for the Phase 1 optimization setup.

5.3.4 Chosen optimal solution from Pareto front and Pareto set

From the Pareto front Figure 5.3, the chosen set of design variables is picked. That set of design variables corresponds to row No. 23 from the Appendix Pareto set Table N.2. This optimal design option results in the lowest 5th eigenfrequency which was minimized for the Phase 0 optimization setup. Even though the No. 23 set of design variables generates the highest 1st eigenfrequency, it is a wise choice to select it, since the optimal values for the 1st eigenfrequency vary much less than for the 5th one.

5.4 Phase 1 optimization setup - Maximization of surface averaged stress FRF bandwidth and area above threshold

5.4.1 Simulation configurations with objective functions

For the Phase 1 optimization setup, the objective functions which are chosen for maximization are the sum of bandwidths frequency ranges, i.e. the difference between upper and lower bandwidth frequencies (sum(FRF BW freq. rng.)), and the sum of areas over a given threshold (sum(FRF A)). For the optimization of FRFs characteristics, the chosen FRF type is the result of the sum of all seven branches' surface averaged normal stress in the longitudinal branch direction. Furthermore, the static load security factor (SF), and the proof mass (PM) consideration (PM Yes/No) are additional parameters which can be varied for each optimization configuration. Also, the two eigenfrequencies to be chosen for Rayleigh damping characterisation are decisive factors which influence FRFs results. The stress FRF thresholds, which are considered for the different optimization configurations, are identified as TH here. They are obtained by using the chosen set of design variables deemed optimal amongst the Pareto set solutions. For all stress FRFs, the threshold value (TH) is estimated with a conversion factor of voltage over stress, see *Theory* section 2.6.

A summary of the Phase 1 simulations configurations is presented in Table 5.2.

Table 5.2: Phase 1 optimization setup simulations configurations.

PM [Y/N]	SF	f_{d1} [Hz]	f_{d2} [Hz]	OF_1	OF_2
N	1.1	f_4	f_5	sum(FRF BW freq. rng) Not. BW TH $\approx 1.79 \cdot 10^6$ [Pa]	sum(FRF A) Not. $\int \Sigma\sigma_{n,i}df$ TH $\approx 1.79 \cdot 10^6$ [Pa]
Y	1.1	f_4	f_5	sum(FRF BW freq. rng) Not. BW TH $\approx 1.79 \cdot 10^6$ [Pa]	sum(FRF A) Not. $\int \Sigma\sigma_{n,i}df$ TH $\approx 1.79 \cdot 10^6$ [Pa]

5.4.2 Input parameters

For the Phase 1 optimization setup, the nominal values of the design variables are taken from the Phase 0 optimization results. These nominal values, and their upper and lower bounds are added in Table 5.3. Note that the number of points (No. Pts) is set to 2, meaning that the sensitivity analysis (SA) prior to the multi-objective optimization will only be executed by varying each design parameter twice from the nominal value. This number of points is low, since the SA is not the main scope of this project but rather the multi-objective optimization. The SA analysis is not of great concern also because the simulation time was minimized as much as possible.

Two different design configurations were prepared for the Phase 1 optimizations. The first design configuration does not have added proof masses placed at the end of the top branches, while the second configuration does. For the second configuration, two new proof mass related input parameters (shown in Table 5.4) are considered in addition to those that are used in the first design configuration (shown in Table 5.3). In Table 5.4, the nominal values used for the two proof mass related design variables are not obtained from the Phase 0 optimization results, but rather set appropriately.

Table 5.3: Phase 1 input parameters used for optimization with maximization of bandwidth and FRF area above threshold

	L [mm]	L_2 [mm]	L_3 [mm]	α [°]	α_3 [°]	w [mm]	w_2 [mm]	w_3 [mm]
Nominal	28.13	24.27	27.72	41.60	59.43	2.716	2.253	2.631
COV	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05
No. Pts	2	2	2	2	2	2	2	2
Dist	N	N	N	N	N	N	N	N
Lower Bnd	10	10	10	20	20	1.5	1.5	1.5
Upper Bnd	35	35	35	80	80	4.0	4.0	4.0
Population Size	see Table 5.5							
No. of generations	see Table 5.5							
Elite Count	4							
Pareto Fraction	1							

Table 5.4: Phase 1 additional input parameters for proof masses (PM) used for optimization with maximization of bandwidth and FRF area above threshold.

	PM ratio from end L_3 [%]	Mass PM [g]
Nominal	10	0.05
COV	0.05	0.05
No. Pts	2	2
Dist	N	N
Lower Bnd	5	0.01
Upper Bnd	30	1.00

The two design configurations optimization simulations for the Phase 1 were run with different GA settings of population size and number of generations (see Table 5.5). There is a risk of the GA converging towards a local minimum area of feasible solutions instead of towards a global minimum solution. A large population size and more numbers of generations is usually preferable, but this results in more GA iterations being performed, and consequently in longer optimization runs. Due to time constrains applicable to this project, and due to the instability issues encountered while using the COMSOL LiveLink for MATLAB feature, a few different combinations of populations size and number of generations were run in Phase 1. For the Phase 2 optimization setups runs, only GA run options 1 and 2 from Table 5.5 were tested.

Table 5.5: Phase 1 and Phase 2 optimization setups simulations GA options, with different population size, number of generations and resulting number of iterations.

	Pop Size	No. Gen	No. Iter.
GA run option 1	25	15	375
GA run option 2	30	25	750
GA run option 3	45	20	900

5.4.3 Pareto fronts

Figure 5.4 shows the Pareto fronts results of Phase 1 optimization, for cases with and without proof masses, and for different population sizes and number of generations. Details about the the Pareto fronts and their corresponding Pareto sets can be found in Appendix O. As can be seen, the optimization runs without proof masses (circles) have converged to smoother Pareto fronts. Conversely, optimization runs with proof masses (asterisks) have converged towards more scattered Pareto fronts values. When comparing the different runs of designs without proof masses, one can also notice that the run with population size 30 and number of generations 25 gave worse optimal results compared to population size 25 and number of generations 15. This confirms the assumption that there is a risk that the optimization procedure might converge towards a local minimum area (of the complete set of feasible solutions) when the population size and number of generations are not large enough. For the runs with proof masses, the scattered Pareto fronts indicate that the GA has not been able to converge towards good solutions as accurately as the runs without proof masses. Even so, including proof masses gives optimal designs which outperform designs without proof masses in both objective functions from Phase 1.

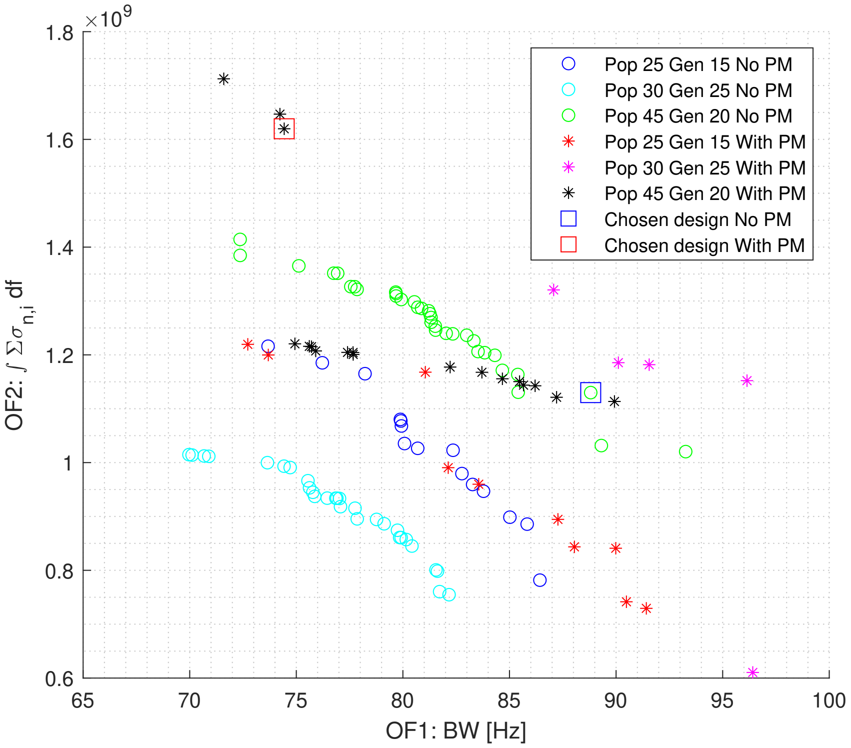


Figure 5.4: Phase 1 Pareto fronts from different optimizations runs of bandwidth (OF1: BW) maximization and FRF area over threshold (OF2: $\int \Sigma\sigma_{n,i}df$) maximization.

5.4.4 Chosen optimal solutions from Pareto fronts and Pareto sets

5.4.4.1 Without proof masses

For the design without proof masses, the Pareto front obtained with a population size of 45 and 20 generations (green circles) is the most accurate Pareto front compared to the other ones. According to Table O.10, the Pareto set shows that the majority of optimal design variables are similar to each other. The favoured optimal solution of design variables is the design associated to a larger sum of bandwidth frequency ranges (OF1). Choosing a design with a very large sum of bandwidth frequency ranges (OF1) over a design with a much bigger area over threshold (OF2) is based on the fact that the area over threshold (OF2) did not vary much. This means that the area over threshold (OF2) did not vary considerably in the y-axis for the current Pareto set designs (black asterisks and green circles) showcased as Pareto fronts in Figure 5.4 .

The design without proof masses is chosen as row No. 22 from the Appendix Pareto set Table O.10, and indicated as a blue square in Figure 5.4.

5.4.4.2 With proof masses

As for the design with proof masses, the Pareto front obtained with a population size of 45 and 20 generations (black asterisks) is the one considered for the optimal design selection. Nonetheless, this Pareto front is not as accurate (more scattered objective functions solutions) as the Pareto front of the same population size and number of generations associated to the design without proof masses. Clearly, as observed in Figure 5.4, the objective function of FRF area over threshold (OF2) varies much more along that chosen Pareto front compared to the Pareto front of a same population size and number of generations associated to the design without proof masses. Generally, according to the Pareto set Table O.12 associated to a population size of 45 and 20 generations, the design parameters vary considerably. Therefore, a design solution favouring a large objective function of area over threshold (OF2) is chosen. The design solution associated to the highest value of the objective function OF2 is neglected because it is still important to consider also a large sum of bandwidth frequency ranges (OF1).

The design with proof masses is chosen as row No. 17 from the Appendix Pareto set Table O.12, and indicated as a red square in Figure 5.4.

5.5 Phase 2 optimization setup - Maximization of surface averaged stress FRFs characteristics and piezoelectric material area coverage consideration

5.5.1 Simulation configurations with objective functions

For the Phase 2 optimization setup, the chosen objective functions for maximization, can be, like for Phase 1, the sum of bandwidths frequency ranges (sum(FRF BW freq. rng.))

and FRF area over a given threshold (sum(FRF A)). An additional objective function, which has to be considered for the optimisation of power output maximisation, would be an objective function taking into account a branch surface variation. As mentioned earlier in the *Methodology* section, for Phase 2 the piezoelectric material surface coverage per branch is varied for each optimization iteration.

An optimal design is desired to have an optimal branch piezoelectric material coverage surface which ensures maximal electrical power output generated by the strained piezoelectric material. To do so, the piezoelectric material coverage surface must be maximised while still ensuring that the surface averaged longitudinal stress is not too low. To achieve that, another objective function takes into account each individual branch area (A_i) multiplied by its respective surface averaged longitudinal stress FRF value (sum(FRF.*A_i A)). Like for the Phase 1, for the optimization of Phase 2 FRFs characteristics, the chosen FRF type is the result of the sum of all seven branches' surface averaged normal stress in the longitudinal branch direction. Now the static load security factor (SF) is not useful since there is no static load analysis performed for this Phase where the tree geometry and proof masses coverage surfaces and mass are already fixed. The proof mass consideration (PM with Yes/No) is still a parameter which can be varied for each optimization procedure configuration. The two eigenfrequencies chosen for the Rayleigh damping characterisation are the same as for the Phase 0 and Phase 1 optimization setups.

Finally, for Phase 2, the geometry design variables are taken from two designs (with and without proof masses) that are chosen from the results of the Phase 1 optimizations. A summary of the Phase 2 simulations configurations is presented in Table 5.6.

Table 5.6: Phase 2 optimization setup simulations configurations.

PM [Y/N]	f_{d1} [Hz]	f_{d2} [Hz]	OF_1	OF_2

N	f_4	f_5	sum(FRF BW freq. rng.)
			Not. BW
			TH $\approx 2.68 \cdot 10^6$ [Pa]
				sum(FRF.*A_i A)
				Not. $\int \Sigma\sigma_{n,i} A_i df$
				TH=0 [N]

N	f_4	f_5	sum(FRF A)
			Not. $\int \Sigma\sigma_{n,i} df$
			TH $\approx 2.68 \cdot 10^6$ [Pa]
				sum(FRF.*A_i A)
				Not. $\int \Sigma\sigma_{n,i} A_i df$
				TH=0 [N]

Y	f_4	f_5	sum(FRF BW freq. rng.)
			Not. BW
			TH $\approx 2.11 \cdot 10^6$ [Pa]
				sum(FRF.*A_i A)
				Not. $\int \Sigma\sigma_{n,i} A_i df$
				TH=0 [N]

Y	f_4	f_5	sum(FRF A)
			Not. $\int \Sigma\sigma_{n,i} df$
			TH $\approx 2.11 \cdot 10^6$ [Pa]
				sum(FRF.*A_i A)
				Not. $\int \Sigma\sigma_{n,i} A_i df$
				TH=0 [N]

5.5.2 Input parameters

In the Phase 2 optimization setup, the design parameters are the upper and a lower bound ratio values (*ub* and *lb*) which define the partial piezoelectric material surface coverage for each branch section (see Figure 3.4). Initial runs prior to Phase 2, showed that the maximum von Mises stress occurs at corners of the tree geometry, except for the first bottom branch (trunk) where stress is maximal at the root point (see Figure 1.2).

Therefore, only one of the two ratios, either upper or lower bound ratio, is needed to be varied for each branch piezoelectric material surface coverage. The location of maximal stress for both chosen design configurations, with and without proof masses, can be seen respectively in Figures 5.10 and 5.9. In total, there are seven design parameters that are used in the Phase 2 optimization configurations, i.e. either upper or lower bound ratio for each one of the seven branches. Their nominal values were set to 0.8, meaning that the piezoelectric material covers 80 % of each branch section starting from the point of maximum stress. The GA lower and upper bounds of the design parameters were set to 0.2 and 1.0 respectively, i.e. the piezoelectric material area coverage can vary from 20 % to 100 % on each branch section. As in the Phase 1 optimization setup, there is no emphasis put on retrieving results from the SA prior to the multi-objective optimization, and therefore the SA simulations duration was minimized.

5.5.3 Pareto fronts

The Phase 2 configurations were optimized for two different pairs of objective functions with a new objective function introduced to quantify the piezoelectric material area coverage (sum(FRF.*A_i A)). Figures 5.5 and 5.6 show the results of Phase 2 optimization, for cases with and without proof masses, and for different population sizes and number of generations. Figure 5.5 shows Pareto fronts associated to the sum of bandwidth frequency ranges (BW) with the new objective function quantifying the piezoelectric material area coverage ($\int \Sigma \sigma_{n,i} A_i df$). Figure 5.6 shows Pareto fronts of the objective function consisting of optimizing stress FRF area above threshold ($\int \Sigma \sigma_{n,i} df$) with also the new objective function quantifying the piezoelectric material area coverage ($\int \Sigma \sigma_{n,i} A_i df$). More details about the the Pareto fronts and their corresponding Pareto sets can be found in Appendix P.

It is apparent that the Pareto fronts of the first pair of objective functions in Figure 5.5 are scattered compared to the Pareto fronts of the second pair of objective functions in Figure 5.6, which are smoother. The sum of bandwidths (OF1) for the different points in Figure 5.5 does not have any large variation, since it varies between 72 Hz and 79 Hz. In other words, varying the piezoelectric material coverage does not seem to vary much the stress FRFs sum of bandwidth frequency ranges, and therefore having it as an objective function might not be relevant for making the choice of an optimal design.

In Figure 5.6 the GA converged towards smoother Pareto curves. In the case without added proof masses, the population size and number of generations does not show any significant difference in the position of the Pareto fronts. In the case with added proof mass, there are some differences in the position of the Pareto fronts. Surprisingly, the Pareto front obtained with a population size 25 and number of generations 15 actually gave better results, even though runs with larger populations and number of generations were tested. Note that due to some technical issues and instability of running the optimization with a larger population size and number of generations (30 and 25 respectively), two runs of the same population size and number of generations were executed for a design with proof masses. The Pareto fronts results for these two identical runs show that they are slightly different since the GA will not always converge to identical results.

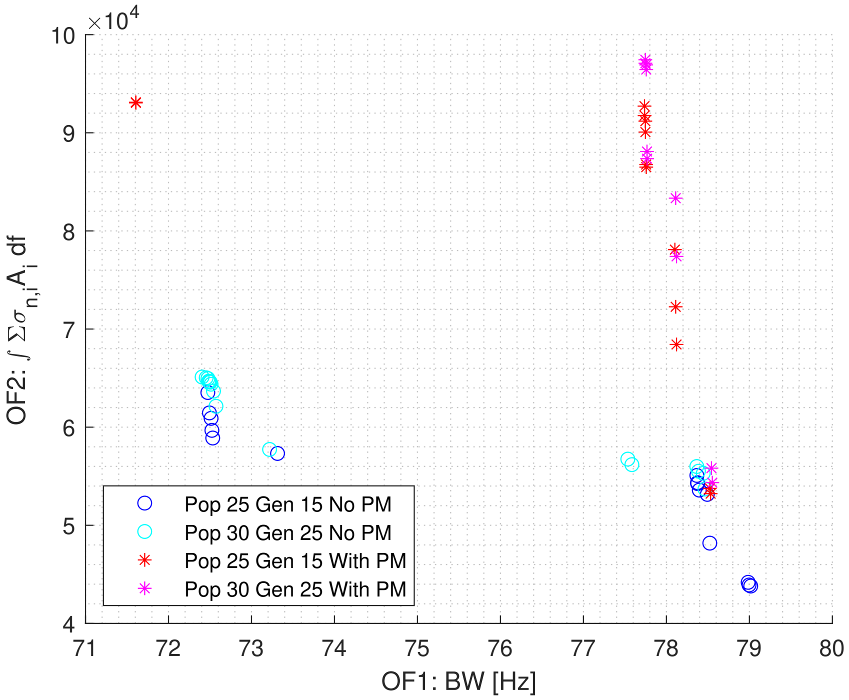


Figure 5.5: Phase 2 Pareto fronts from different optimizations runs of bandwidth (OF1: BW) maximization and FRF area over threshold (OF2: $\int \Sigma\sigma_{n,i}A_i df$) maximization.

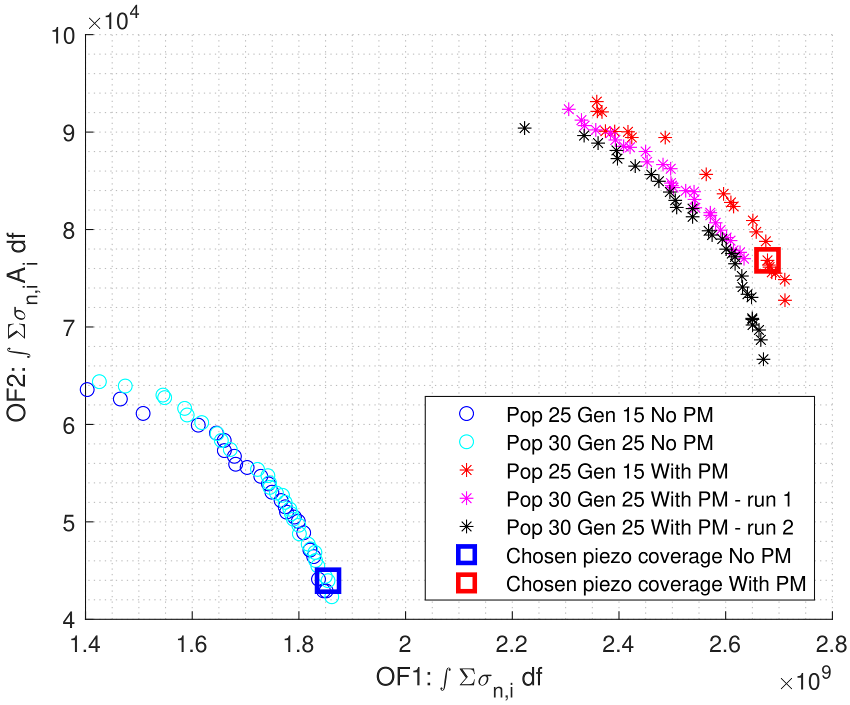


Figure 5.6: Phase 2 Pareto fronts from different optimizations runs of FRF area over threshold (OF1: $\int \Sigma\sigma_{n,i}df$) maximization and FRF area without threshold (OF2: $\int \Sigma\sigma_{n,i}A_i df$) maximization.

Since the Pareto fronts of the second pair of objective functions in Figure 5.6 are smoother than the Pareto fronts of the first pair of objective functions seen in Figure 5.5, the second pair of objective functions seems to be more relevant for selecting an optimal piezoelectric material coverage. It was therefore decided to select from these Pareto sets a piezoelectric material coverage that could be suitable. Judging from the Pareto sets Tables found in Appendix P, it is clear that the results of the piezoelectric material coverage proportions (*lb* and *ub*) do not always converge to be symmetric. The symmetry between branches is here defined when they mirror each other through a vertical symmetry midline at $x = 0$ in the global coordinate system (see Figure 1.2). Nevertheless, few results are having some symmetry, and it would be best to pick optimal designs which are as much symmetric as possible. Two piezoelectric material coverage design configurations are selected, one with added proof masses and one without. Both of these piezoelectric material coverage configurations are associated to Pareto fronts designs that prioritize maximizing OF1 over OF2 (see Figure 5.6).

5.5.4 Chosen optimal solutions from Pareto fronts and Pareto sets

5.5.4.1 Without proof masses

For the configuration of the fractal design with no added proof masses, the chosen piezoelectric material coverage proportions (*lb* and *ub*) are picked from row No. 6 of the Pareto set Table P.9 shown in the Appendix. This design solution is associated to a population size of 30 and a number of 25 generations, and is indicated in Figure 5.6 with a blue square.

5.5.4.2 With proof masses

For the configuration of the fractal design with added proof masses, the chosen piezoelectric material coverage proportions (*lb* and *ub*) are picked from row No. 19 of the Pareto set Table P.3 shown in the Appendix. This design solution is associated to a population size of 25 and a number of 15 generations, and is indicated in Figure 5.6 with a red square.

5.6 Chosen optimal designs of fractal energy harvesters from Pareto fronts and Pareto sets

Table 5.7 gives a summary of the final geometry designs for the fractal energy harvesters that are chosen according to the Phase 1 optimization results. Additionally in Table 5.7, the piezoelectric material surface coverage proportions for these fractal energy harvesters are also specified according to the Phase 2 optimization results. This Table also shows the population size and number of generations of the Pareto fronts and sets which are used to select these optimal design parameters, and the values of the objective functions which they are associated to.

Table 5.7: Final design parameters for Phase 1 and Phase 2 optimal solutions.

Phase	Pareto Set & Front sol.	Without PM	With PM
1	L [mm]	25.51	22.65
	L_2 [mm]	17.03	26.20
	L_3 [mm]	31.17	20.57
	α [°]	51.14	57.42
	α_3 [°]	62.94	45.41
	w [mm]	3.060	2.934
	w_2 [mm]	1.746	3.380
	w_3 [mm]	2.995	1.919
	PM ratio L_3 [%]	-	25.41
	PM mass [g]	-	0.2385
GA sol.	Pop	45	45
	Gen	20	20
	OF1 [Hz]	88.81	74.43
	OF2	$1.130 \cdot 10^9$	$1.620 \cdot 10^9$
2	lb1 [%]	-	-
	ub1 [%]	26.37	31.63
	lb2 [%]	63.38	-
	ub2 [%]	-	76.87
	lb3 [%]	65.18	-
	ub3 [%]	-	79.82
	lb4 [%]	-	-
	ub4 [%]	42.69	45.33
	lb5 [%]	-	-
	ub5 [%]	24.76	64.53
	lb6 [%]	-	-
	ub6 [%]	28.18	62.05
	lb7 [%]	-	-
	ub7 [%]	42.73	52.76
GA sol.	Pop	30	25
	Gen	25	15
	OF1	$1.855 \cdot 10^9$	$2.678 \cdot 10^9$
	OF2	$4.396 \cdot 10^4$	$7.683 \cdot 10^4$

5.6.1 Final design of Phase 1 optimization

Figures 5.7 and 5.8 show the two overall geometry designs of the piezoelectric energy harvester that are chosen from the Phase 1 optimization results, without and with proof masses respectively. From these Figures it is clear that the geometries are quite different.

5.6.1.1 Without proof masses

In the configuration with no added proof masses (see Figure 5.7), the two mid branches extending from the trunk are quite short and thin compared to the other branches. The two top pairs of branches that continue from the mid branches are much larger in size.

This implies that these top branches must have more mass compared to their parent branches, meaning that the Genetic Algorithm converged to a geometry that mimics a design with added proof masses on the top branches. Due to their small size, the two mid branches seem to be acting more as connectors between the trunk and the top branches. Additionally, having the free ends for the top branches in the middle close to each other suggests that there is a high risk of contact between these parts of the fractal energy harvester. That is particularly risky, since this contact could occur in a real application of the energy harvester where a lateral motion is possible.

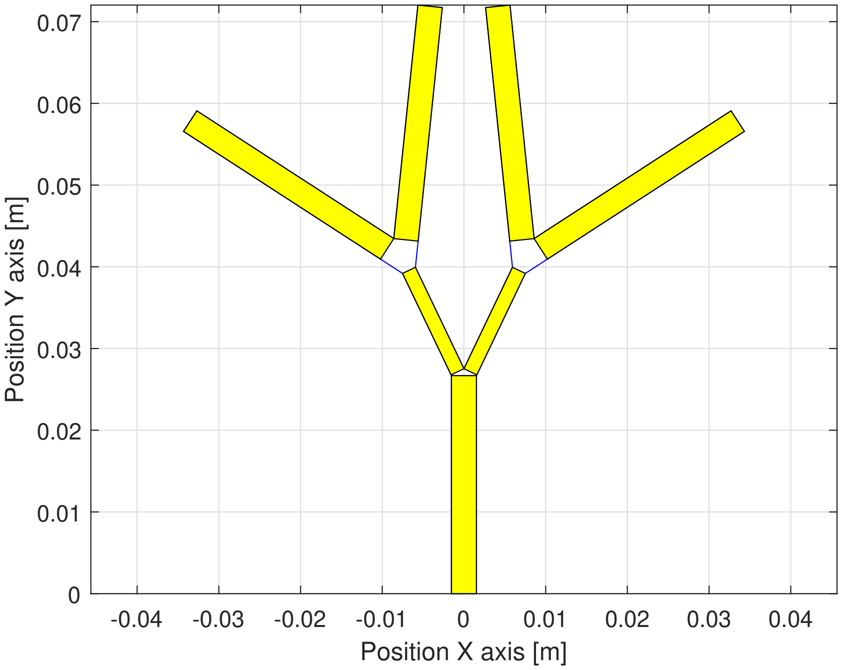


Figure 5.7: Phase 1 final design geometry of the fractal energy harvester without added proof masses and with full piezoelectric material branch coverage (yellow areas).

5.6.1.2 With proof masses

In the chosen design configuration with added proof masses (see Figure 5.8), the top branches of the fractal design are much smaller compared to the trunk and mid branches. Here the mid branches are also much longer and wider than they are for the design without proof masses. An important aspect to consider is that the free ends of the design are much further away from the symmetry line at $x = 0$ compared to the design without proof masses. Unlike the case for the design without proof masses, there is most certainly no risk of contact between the middle top branches due to a lateral dynamic or static applied load.

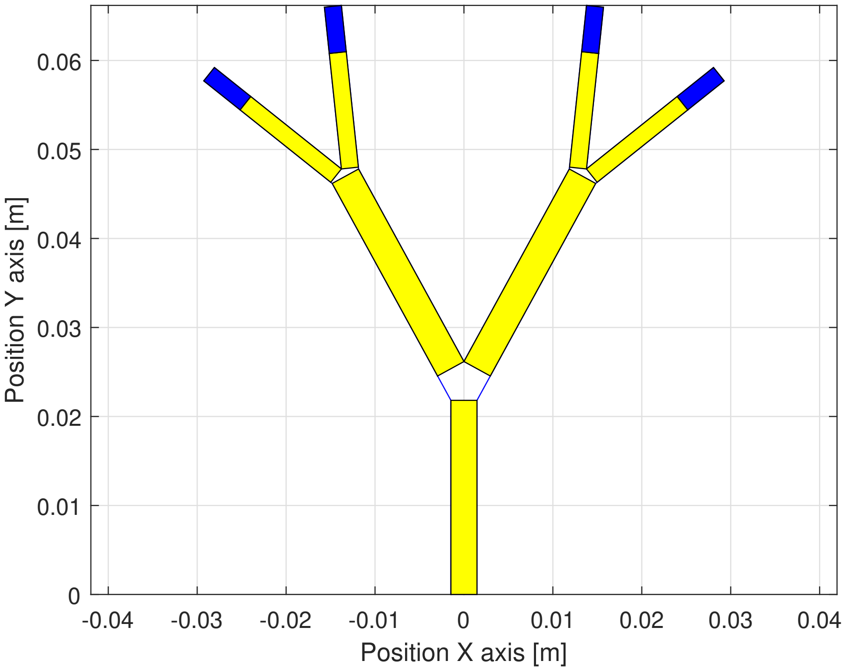


Figure 5.8: Phase 1 final design geometry of the fractal energy harvester with added proof masses (blue areas) and with full piezoelectric material branch coverage (yellow areas).

5.6.2 Final design of Phase 2 optimization

In Figures 5.9 and 5.10 the results for Phase 2 of the piezoelectric material coverage are shown for the chosen fractal harvester designs, without proof masses and with proof masses respectively. The red cross marks indicate the points of highest von Mises stress for each branch at a specific reference frequency. Before the Phase 2 optimization simulations were carried out, these points of maximum stress were computed and they seem to be located at the corners of each branch and at the trunk root. As expected, there is a stress concentration at the corners. Consequently, only one limit of piezoelectric material coverage proportion (*lb* or *ub*) is varied for each branch section.

5.6.2.1 Without proof masses

For the configuration without proof masses (see Figure 5.9), the mid branches have a large coverage proportion of piezoelectric material, while the trunk and the top branches have a smaller coverage proportion.

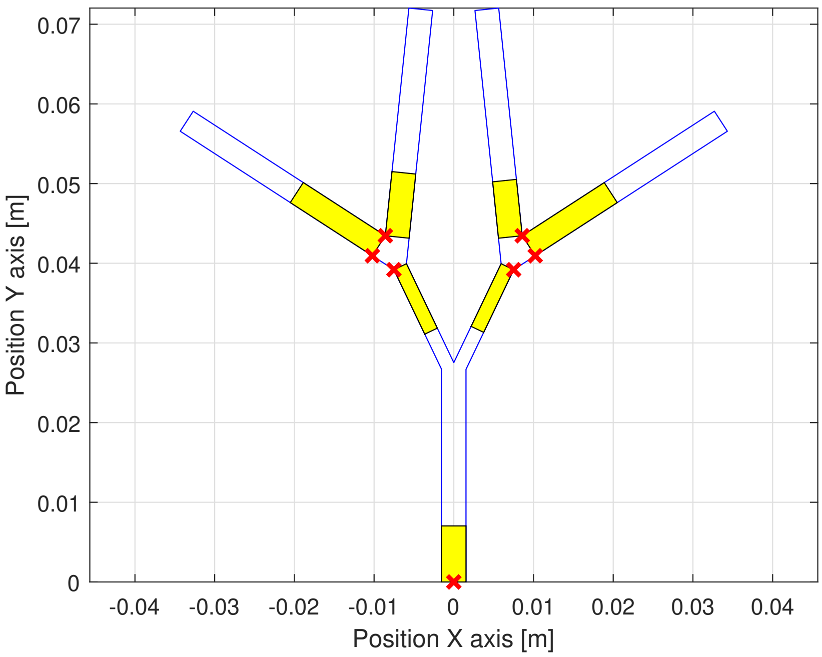


Figure 5.9: Phase 2 final design geometry of the fractal energy harvester without added proof masses and with partial piezoelectric material branch coverage (yellow areas). Red cross marks indicate the points of maximum von Mises stress for each branch.

5.6.2.2 With proof masses

In the design configuration with added proof masses (see Figure 5.10), some of the maximum stress points are now shared between two neighbouring branches. As with the design without proof masses, the mid branches have a large piezoelectric material coverage proportion. On the contrary, the trunk branch has a smaller piezoelectric material coverage proportion just like it is the case for the design without proof masses.

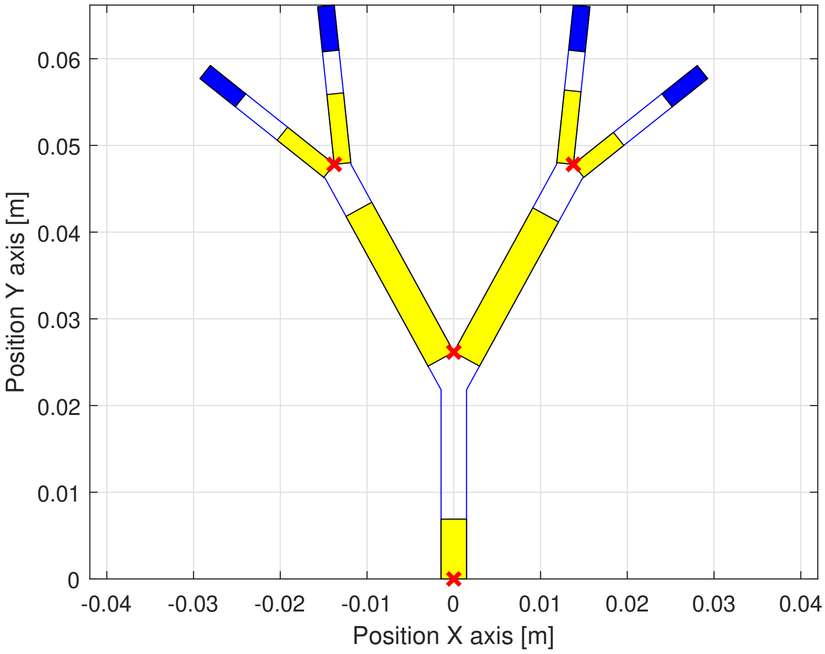


Figure 5.10: Phase 2 final design geometry of the fractal energy harvester with added proof masses (blue areas) and with partial piezoelectric material branch coverage (yellow areas). Red cross marks indicate the points of maximum von Mises stress for each branch.

5.7 Modal analysis and static load case

Table 5.8 displays the eigenfrequencies of the two chosen geometry designs of the fractal energy harvester, as well as the maximum von Mises stress and the maximum free end displacement at the top branches for the static analysis.

The 1^{st}, 2^{nd} and 5^{th} eigenfrequencies are similar between the two design configurations (with and without proof masses), while the 3^{rd} and 4^{th} are much more different between the two design configurations. The 4^{th} and 5^{th} eigenfrequencies are much further apart from each other in the configuration without added proof mass compared to the configuration with added proof masses. These are the two eigenfrequenices which are relevant for generating a high voltage output, and if they are close enough to each other they produce a large FRF bandwidth. This means that the design with proof masses produces better FRF characteristics, which are a larger bandwidth and a greater area above the threshold.

The maximum von Mises stress can be said to be, for both design configurations, in a safe margin smaller than the material yield strength of 240 MPa divided by a safety factor of 1.1, which gives approximately a stress limit of 218 MPa (σ_Y/SF). Figure 5.11 shows, for both design configurations, the von Mises stress distribution for the static analysis of the 10 g-force load case. As expected, it confirms that the maximum static stress is located at the root point (0,0) of the trunk for both design configurations.

Additionally, for both design configurations, the maximum static free end displacement at the top branches is found to be slightly more than one centimeter.

Table 5.8: Eigenfrequencies (f_i), maximum von Mises stress (σ_{vM}^{max}) and maximum free end displacement at the top branches ($disp.^{max}$), for Phase 1 configurations with and without proof masses. Piezoelectric material mass, stiffness and added damping is neglected

	f_1	f_2	f_3	f_4	f_5	σ_{vM}^{max}	$disp.^{max}$
	[Hz]	[Hz]	[Hz]	[Hz]	[Hz]	[MPa]	[mm]
Without PM	18.185	39.208	91.295	109.690	171.41	182.02	11.3
With PM	17.771	37.581	153.120	161.920	179.690	204.02	10.5

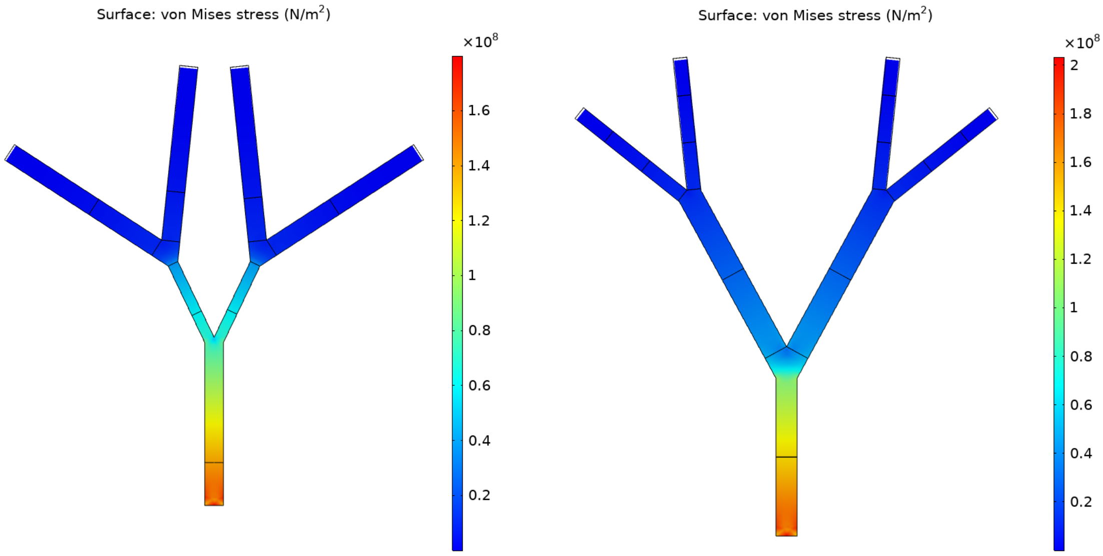


Figure 5.11: Von Mises stress distribution for the static load case (10 g-force) for the two fractal tree geometry designs, without proof masses (left) and with added proof masses (right).

5.7.1 Shapes of 4^{th} and 5^{th} eigenmodes

Figures 5.12 and 5.13 show the shapes of the 4^{th} and 5^{th} eigenmodes for the two chosen fractal energy harvester designs. These modes are of most interest since they are the main contributors to the total stress FRF output, and therefore to the total voltage FRF output as well (see following section about the *Dynamic load case - FRF results*). As a reminder, the 4^{th} and 5^{th} eigenfrequencies are also the frequencies used for modeling the Rayleigh damping. From these Figures it is clear that these eigenmodes have different shapes and that they are different depending on the geometry of the fractal energy harvester.

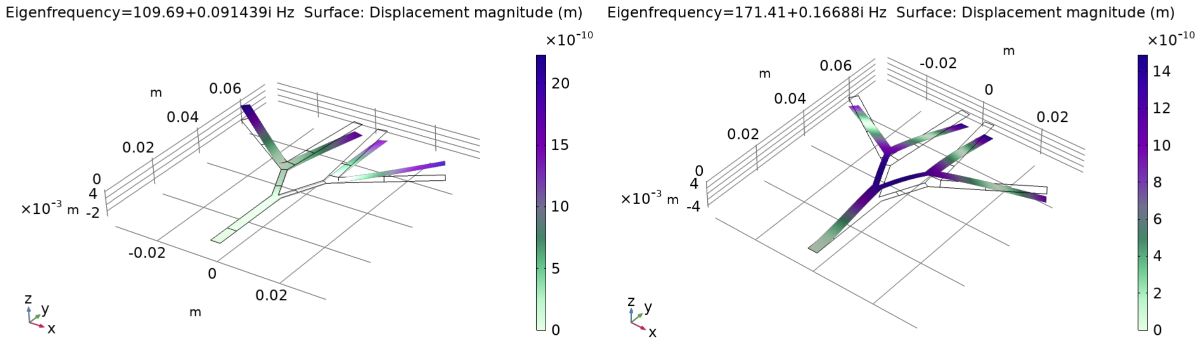

Eigenfrequency=109.69+0.091439i Hz Surface: Displacement magnitude (m)
×10⁻¹⁰
20
15
10
5
0
m
0.06
0.04
0.02
0
4
-2
×10⁻³ m
-0.02
0
0.02
m
z
y
x
Eigenfrequency=171.41+0.16688i Hz Surface: Displacement magnitude (m)
×10⁻¹⁰
14
12
10
8
6
4
2
0
m
0.06
-0.02
0
0.02
0.04
0.02
0
4
-4
×10⁻³ m
m
z
y
x

Figure 5.12: Shapes of 4^{th} (left) and 5^{th} (right) eigenmodes of the fractal energy harvester design without added proof masses.

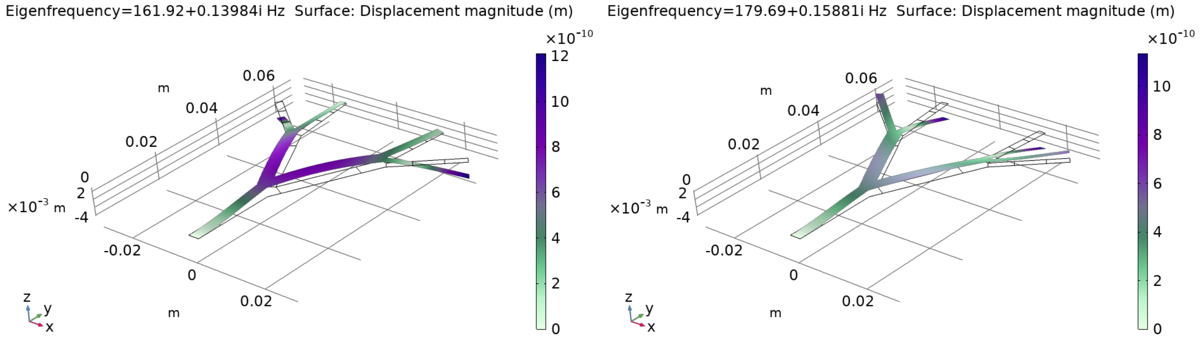

Eigenfrequency=161.92+0.13984i Hz Surface: Displacement magnitude (m)
×10⁻¹⁰
12
10
8
6
4
2
0
m
0.06
0.04
0.02
0
2
-4
×10⁻³ m
-0.02
0
0.02
m
z
y
x
Eigenfrequency=179.69+0.15881i Hz Surface: Displacement magnitude (m)
×10⁻¹⁰
10
8
6
4
2
0
m
0.06
0.04
0.02
0
2
-4
×10⁻³ m
-0.02
0
0.02
m
z
y
x

Figure 5.13: Shapes of 4^{th} (left) and 5^{th} (right) eigenmodes of the fractal energy harvester design with added proof masses.

5.8 Dynamic load case - FRF results

Post-processed results are here presented as total surface averaged voltage and stress FRFs. The FRFs are showing the cumulative value of the surface average by considering the contribution of all seven branches.

It cannot be determined by relying on the total surface averaged von Mises stress FRF, if the structure would yield or not. This cannot be done simply because it is a cumulative stress FRF result and a surface average value which does not represent the actual maximal stress located at fractal tree corners and trunk root. On the other hand, the FE models had approximated damping ratios which render also the amplitudes of all FRFs peaks to be approximated.

5.8.1 Total surface averaged voltage outputs

The total voltage output for the two fractal energy harvesters designs, when adding the surface averaged voltage outputs from each branch, is presented in Figure 5.14. It shows the total voltage FRFs outputs computed by FE models using multi-layered shells where piezoelectric material is placed on top of the structural steel frame. It is clear here that the 4^{th} mode of the design without proof masses (blue lines) gives a much lower output than the design with proof masses (red lines). The Phase 2 optimization designs (dotted lines) produce higher surface averaged voltage FRFs outputs compared to the Phase 1 designs (full lines), specially at frequencies close to the eigenfrequencies, which are distinguishable as FRFs peaks. This makes sense since a partial piezoelectric material surface coverage per branch generates higher surface averaged FRFs due to higher stress concentration at the corners and at the trunk root.

If for Phase 2, the only optimized objective function would have been the maximization of OF1 ($\int \Sigma \sigma_{n,i} df$), the GA would have most likely suggested a single optimal design solution characterised by very small piezoelectric material coverage surfaces concentrated close to stress concentration points.

Finally, the last noticeable difference between Phase 1 and Phase 2 total voltage FRFs are the eigenfrequencies shifts related to FRFs peaks shifts. This can be explained by the fact that the mass and stiffness properties of the piezoelectric material are not neglected within the FE models used for voltage FRFs computation. Inevitably, a difference of piezoelectric material surface coverage between Phase 1 and Phase 2 generates a shift of voltage FRFs peaks. As proven by voltage FRFs results, the difference of surface coverage between Phase 1 and Phase 2 is obviously greater for the design without added proof masses. One would be able to observe this difference by looking at Figures 5.8 and 5.7 for the Phase 1 optimal designs respectively with and without proof masses, and then at Figures 5.10 and 5.9 for the Phase 2.

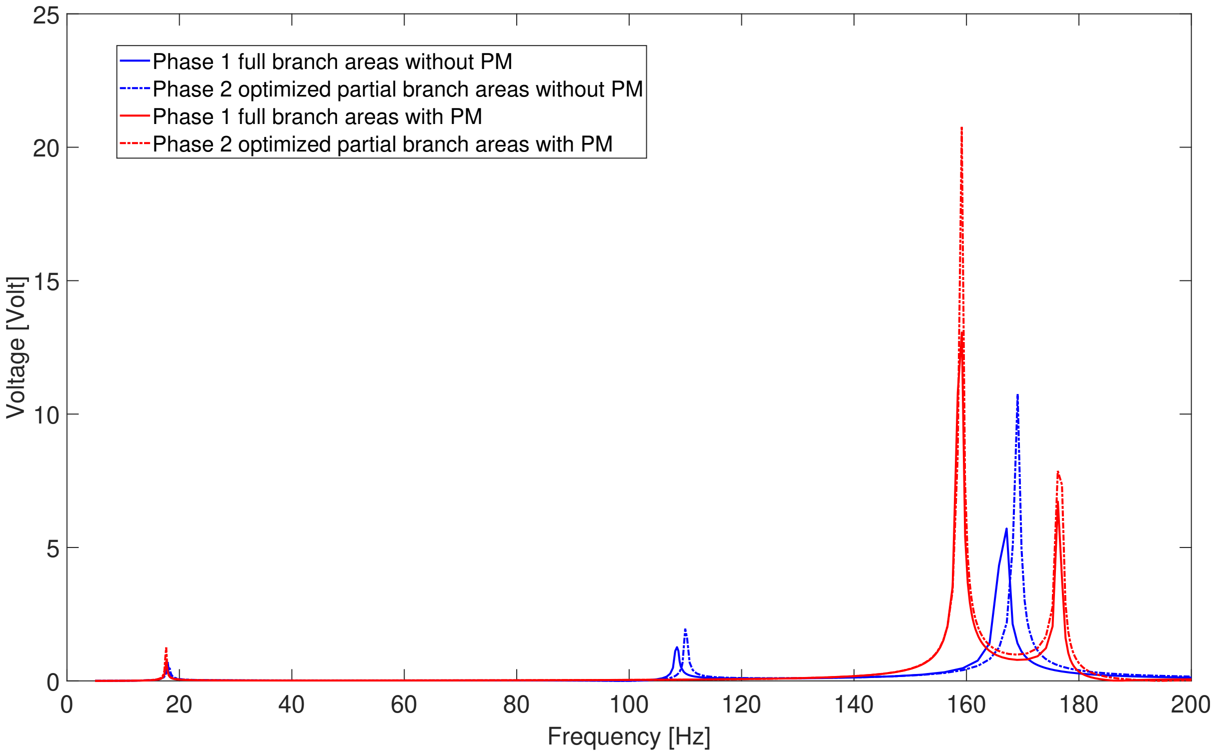


Figure 5.14: Comparison of total surface averaged voltage output FRF, with and without proof masses (PM), between Phase 1 and 2.

5.8.2 Total surface averaged longitudinal stress outputs

Figure 5.15 shows, for the two optimal fractal energy harvesters designs, the total surface averaged stress FRF outputs when adding the surface average stress contribution from each branch. FE models used for stress FRF analysis do not consider the added mass and stiffness due to the piezoelectric material surface coverage. Therefore, even though there is a difference of piezoelectric surface coverage between Phase 1 and Phase 2 designs, there is no shift in stress FRFs peaks and eigenfrequencies between their results.

As in the case of voltage FRFs, the Phase 2 optimization designs (dotted lines) give higher surface averaged stress FRFs outputs compared to the Phase 1 designs (full lines). Clearly, a partial piezoelectric material surface coverage per branch generates higher surface averaged FRFs than a total surface coverage.

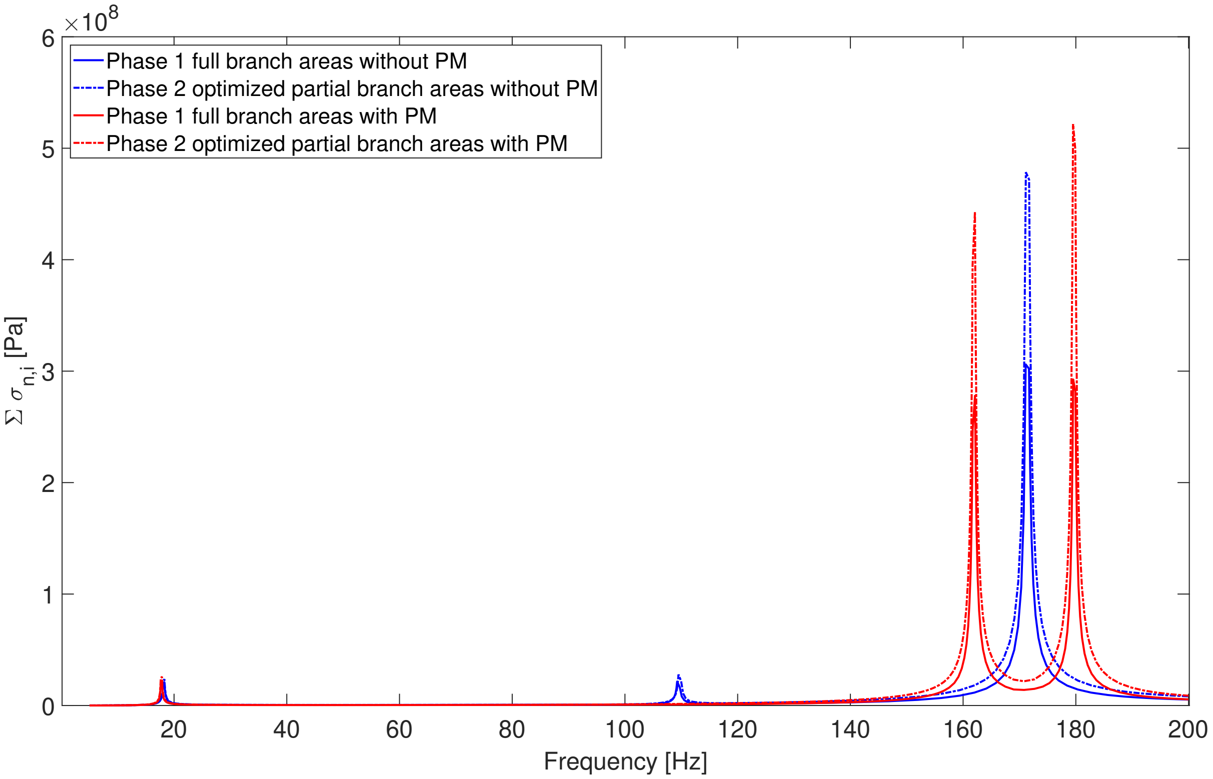


Figure 5.15: Comparison of total surface averaged longitudinal stress ($\Sigma\sigma_{n,i}$) output FRF, with and without proof masses (PM), between Phase 1 and 2.

5.8.3 Total surface averaged longitudinal stress with consideration of piezoelectric material coverage area outputs

Figure 5.16 shows FRFs of the sum of all branches' surface averaged longitudinal stress multiplied by the piezoelectric material branch coverage area. This Figure presents FRFs of Phase 1, with total piezoelectric material branch surface coverage, and compares them with FRFs from Phase 2 with partial piezoelectric material branch surface coverage. All these FRFs represent the sum of all branches' averaged normal force in the longitudinal direction. This cumulative longitudinal force relates to the total power extraction, since the piezoelectric material coverage surfaces influence not only the total surface averaged voltage but also the total electric current generation.

From Figure 5.16, it appears that the FRF values from Phase 1 for all frequency samples are higher than the FRF values from Phase 2. This implies that a full piezoelectric material surface coverage for each branch would generate more power output than a partial coverage. It was expected however that a partial coverage of piezoelectric material for each branch would generate more power output because it would avoid covering surfaces where "S" deformations occur for any individual branch.

In order for "S" deformations, that occur for a specific branch at eigenmodes above 200 Hz, to be taken into account, the FRF analysis would have to be performed for higher frequencies. In the *Methodology* section, Figure 3.1 presents a typical example of a design without proof masses where an "S" deformation within the trunk is encountered at 500 Hz. In the current frequency range there are no modes which produce "S" deformations within any individual branch, but rather modes comparable to a cantilever beam's 1^{st} mode (see 4^{th} and 5^{th} eigenmodes Figures 5.12 and 5.13). This analogy between fractal energy harvester modes and a cantilever beam's modes is good for visualizing how a total branch surface coverage by piezoelectric material could generate voltage cancellation for higher modes having "S" deformations. Finally, a total surface coverage by piezoelectric material for each branch would be better for producing higher power output at frequencies below 200 Hz. Even though that is the case, the FRFs differences between Phase 1 and Phase 2 are only distinguishable at specific eigenfrequencies and nearby.

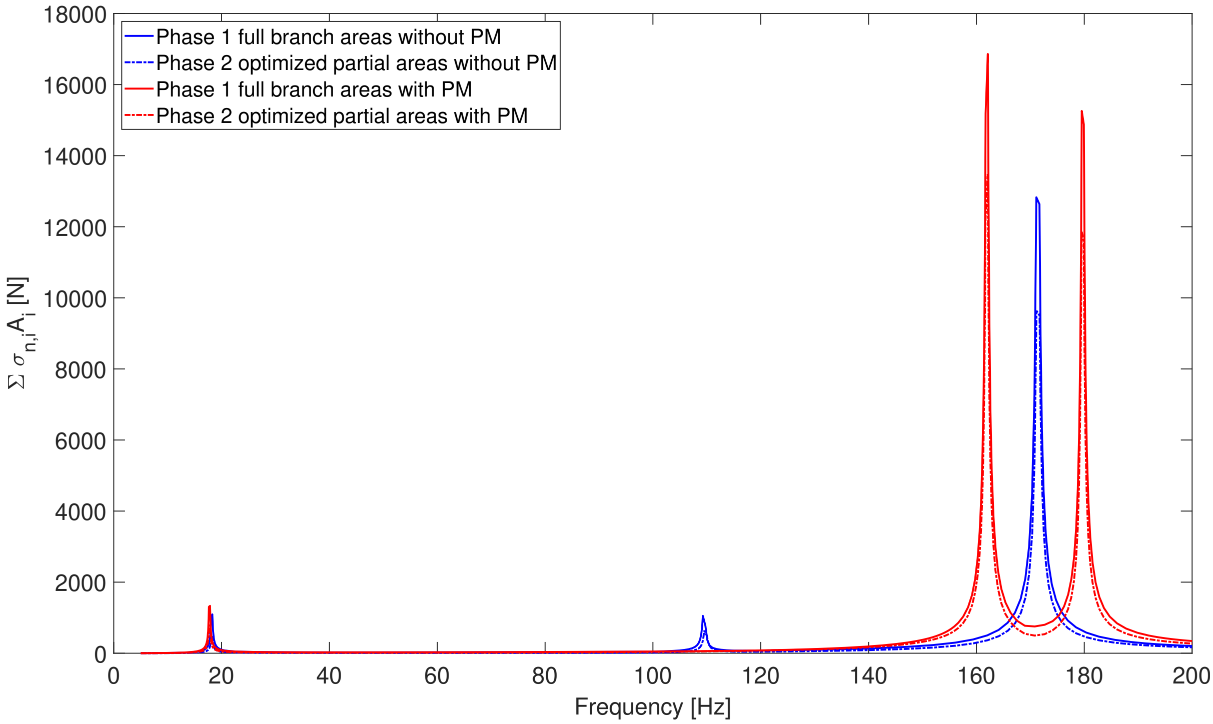


Figure 5.16: Comparison of total surface averaged longitudinal stress times the branch surface which is used for piezoelectric material coverage ($\Sigma\sigma_{n,i}A_i$) output FRF, with and without proof masses (PM), between Phase 1 and 2.

5.8.4 Comparison of total surface averaged voltage and total surface averaged longitudinal stress outputs

Figures 5.17 and 5.18 show respectively for designs with and without proof masses comparisons between voltage and stress FRFs for both Phases 1 and 2.

Both Figures prove that there is not a considerable shift in eigenfrequencies between voltage and stress FRFs. This explains why once again the added mass and stiffness due to the piezoelectric material consideration can be neglected. From Phase 1 to Phase 2, there is a reduction of piezoelectric material surface coverage, resulting in an overall mass reduction but less decrease in stiffness. Due to a structure mass reduction occurring from Phase 1 to Phase 2, the voltage FRFs peaks get shifted to slightly higher eigenfrequencies.

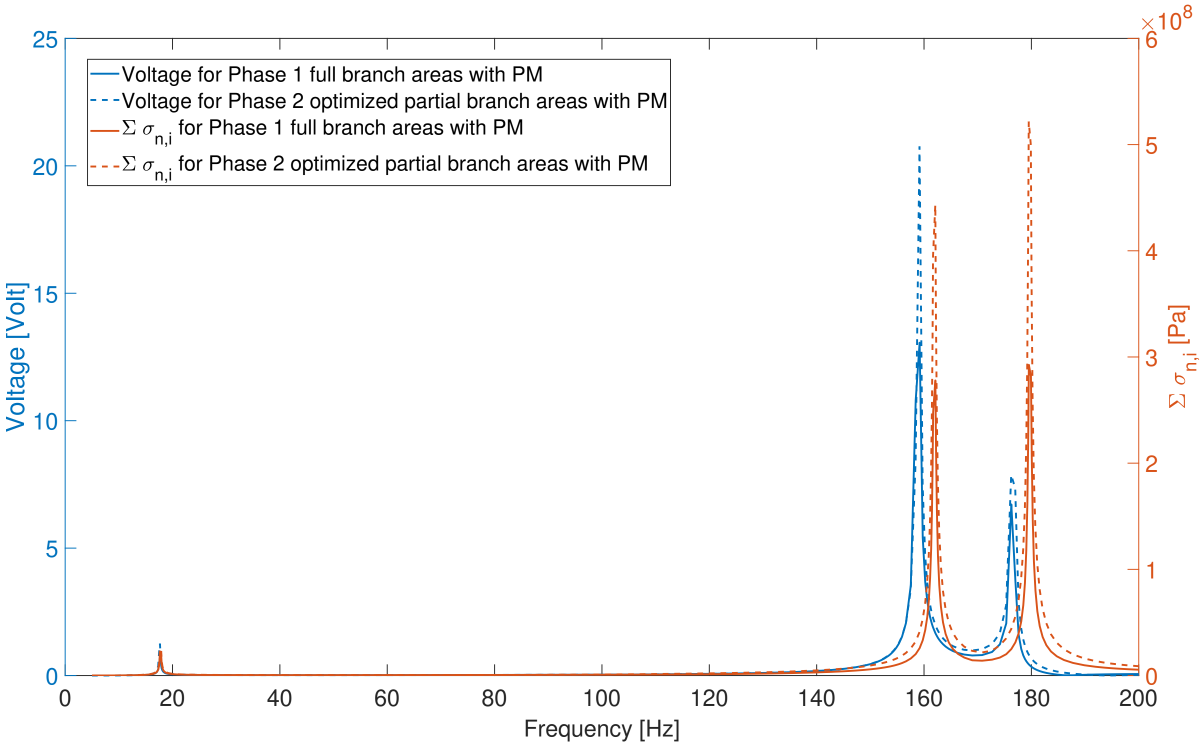


Figure 5.17: Comparison of total surface averaged voltage and total surface averaged longitudinal stress ($\Sigma\sigma_{n,i}$) FRF outputs, with proof masses (PM), between Phase 1 and 2.

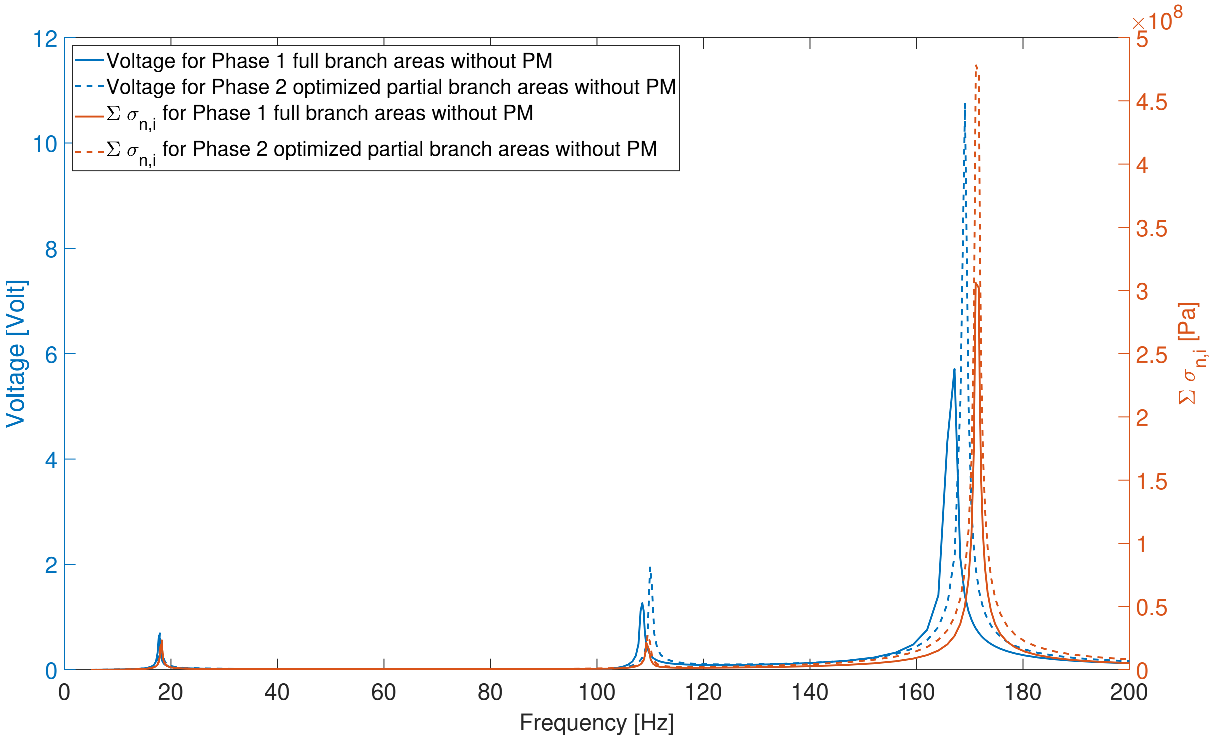


Figure 5.18: Comparison of total surface averaged voltage and total surface averaged longitudinal stress ($\Sigma\sigma_{n,i}$) FRF outputs, without proof masses (PM), between Phase 1 and 2.

5.8.5 Maximum free end displacements at top branches outputs

The FE models used for computing the FRF of maximal displacement at the tip of the top branches are the same FE models as the ones used for the stress FRF analysis. As previously stated, those FE models do not consider added mass and stiffness due to piezoelectric material surface coverage.

Figures 5.19 and 5.20 show the FRFs of maximal displacement at the end of the top branches respectively for the designs with and without proof masses. Obviously the design without proof masses has much lower maximal displacement FRFs amplitudes, approximately 1000 times lower, than the design with proof masses. What differs distinctly for these two Figures compared to previous stress and voltage FRFs Figures, is that the 1^{st} eigenmode produces an FRF output of an amplitude which is comparable with the amplitudes of 4^{th} and 5^{th} eigenmodes.

Like for the stress and voltage FRFs Figures, the dominating eigenfrequencies in terms of FRFs outputs are the 1^{st}, 4^{th} and 5^{th}. Thus, a proper Rayleigh damping model has to consider two of these modes, and the 4^{th} and 5^{th} are the chosen modes because they are closer to each other. Although these two modes are used for modelling the Rayleigh damping, their associated displacement amplitudes are still large in comparison with the 1^{st} eigenfrequency displacement amplitude. This means that even with a damping applied to the 4^{th} and 5^{th} modes, the FRFs outputs near their corresponding eigenfrequencies are large and they are proper modes to be used for the Rayleigh damping.

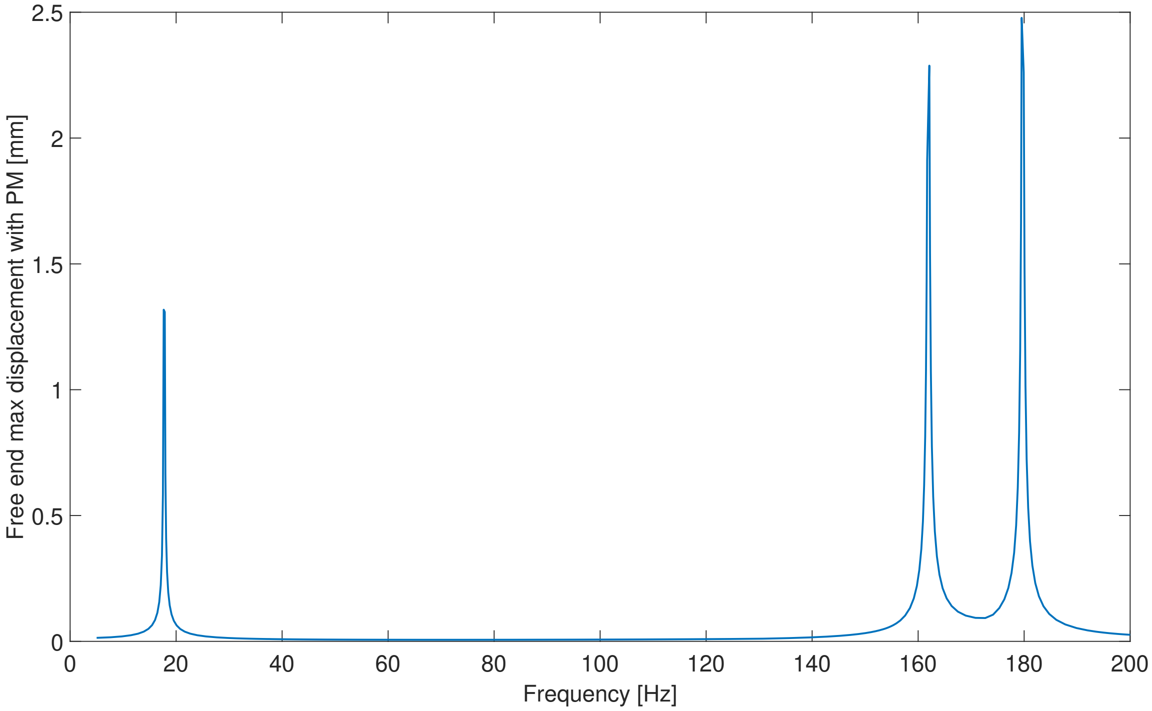


Figure 5.19: Maximum free end displacement at top branches FRF for the chosen fractal energy harvester design with added proof masses (PM). Only the 1^{st}, 4^{th} and 5^{th} modes have a considerable and visible FRF contribution.

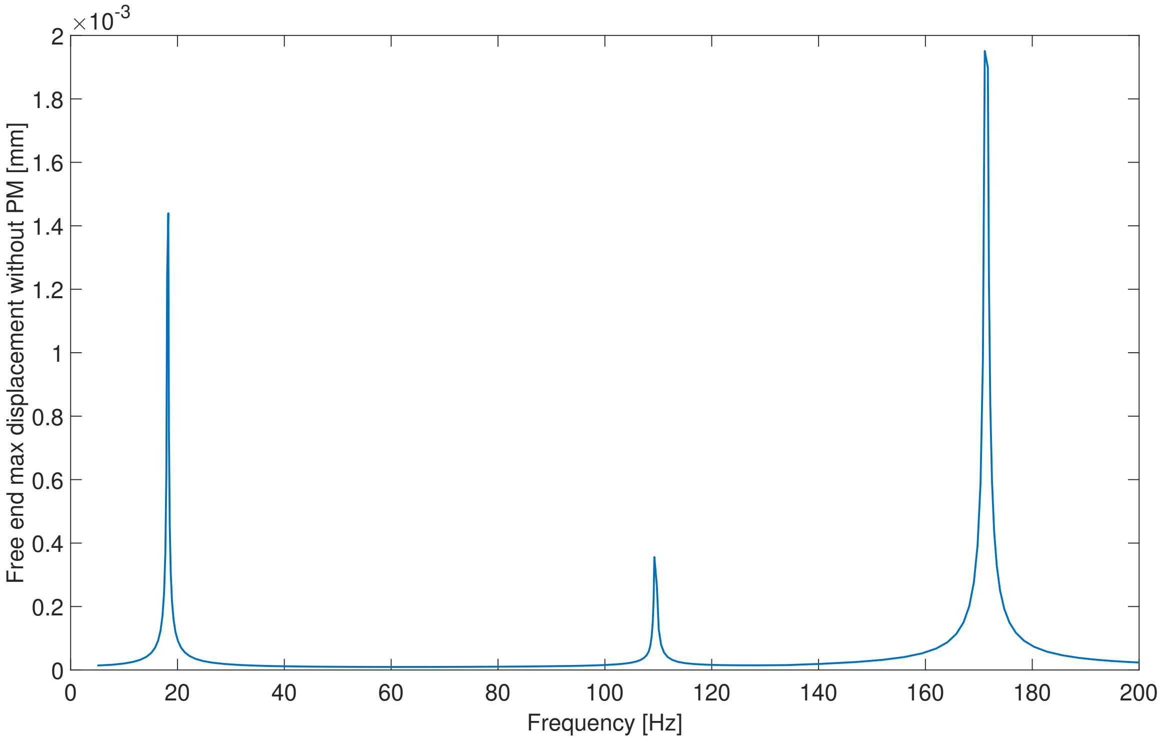


Figure 5.20: Maximum free end displacement at top branches FRF for the chosen fractal energy harvester design without added proof masses (PM). Only the 1^{st}, 4^{th} and 5^{th} modes have a considerable and visible FRF contribution.

5.8.6 Comparison of total surface averaged longitudinal stress and total surface averaged von Mises stress outputs

Figure 5.21 compares total branch longitudinal stress to von Mises stress FRFs outputs, for Phase 2, with and without added proof masses. It reveals that the branch surface averaged von Mises stress FRF output is similar to the longitudinal stress FRF output. The noticeable difference between the von Mises and longitudinal stress outputs, is that the von Mises stress output (dotted lines) gives higher averaged stress values, especially at eigenfrequencies where the FRF peaks occur. Since the von Mises stress does not consider the sign for the stress being present on the branches surfaces, i.e. if there is a tension or compression, there is a risk of having overestimated stress outputs when averaging over a branch surface. By taking into account the von Mises surface averaged stress, there is a risk of ignoring that some parts of each branch section have negative stress components. In conclusion, the preferable stress FRF output to be used for optimization is the total surface averaged longitudinal stress.

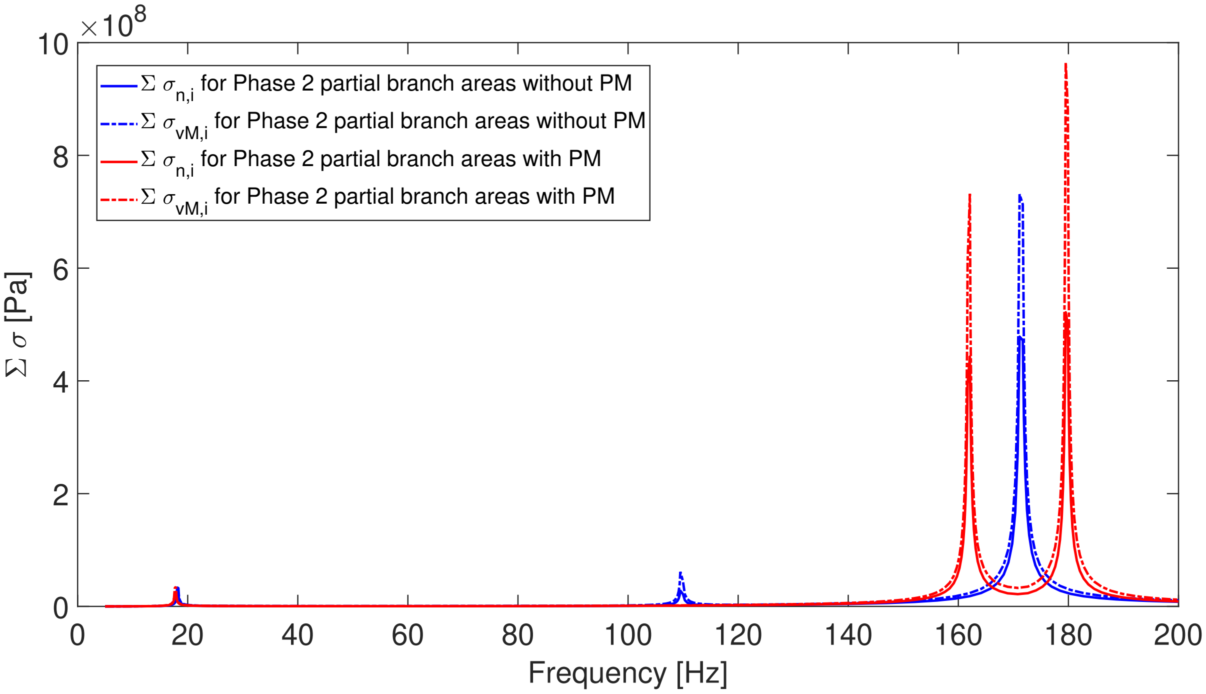


Figure 5.21: Comparison of total surface averaged longitudinal stress ($\Sigma\sigma_{n,i}$) and total surface averaged von Mises stress ($\Sigma\sigma_{vM,i}$) FRF outputs, with and without proof masses (PM), for Phase 2.

5.9 Final chosen design of fractal energy harvester with added proof masses

The final chosen fractal energy harvester design considers the placement of proof masses at the end of top branches with each having a mass of approximately 0.24 g and a specific coverage surface. The design with proof masses has its dimensions displayed in Table 5.7 and its geometry portrayed in Figure 5.10. For a manufactured prototype of this design, the proof masses must be made of a material with a certain density and with proper dimensions. The material density of the proof masses and their dimensions must fit best with the optimal solution.

5.9.1 Reasons why the design with proof masses is preferable

The design with proof masses outperforms the design with no proof mass consideration for many reasons, such as due to higher voltage (see Figure 5.14) and longitudinal stress (see Figure 5.15) FRF outputs. Furthermore, the power generation, which is related to the total surface averaged longitudinal stress times the branch surface (see Figure 5.16), is also greater for the design with proof masses.

The design with proof masses looks overall more robust and capable of withstanding greater static loads with seemingly less stress concentration at the corners, refer to Figure 5.11 displaying the von Mises stress due to static load.

Moreover, the design with proof masses has its 4^{th} and 5^{th} eigenfrequencies much closer to each other than the design without proof masses resulting in a larger bandwidth in that frequency range. The 4^{th} and 5^{th} modes, as explained earlier, are crucial for energy harvesting because they generate the largest bandwidths frequency ranges and the highest values for the longitudinal stress and voltage FRFs outputs.

5.9.2 Drawbacks for the design with proof masses

On the other hand, the design with proof masses has minor disadvantages compared to the design without proof masses, such as higher top branches free ends maximal displacement for a static load (see Table 5.8) or a dynamic load (see Figures 5.19 and 5.20) applied. The design with proof masses has also a higher static von Mises stress at the trunk root location of the fractal tree design. Nonetheless, the design with proof masses is still a better design option, knowing that the only considerable difference between the two design configurations is the top branches free ends maximal displacement when a dynamic load is applied.

5.9.3 Placement of piezoelectric material and space allocation for cable connections

A full branch surface coverage by the piezoelectric material is ideal because it generates more power extraction. It is a better option if possible, because the 4^{th} and 5^{th} eigenmodes for both configurations, with and without proof masses, have no "S" shape deformations within any branch (see Figures 5.13 and 5.12). Such undesired "S" shape deformations are associated to both compression and tension surface stresses and they result in undesired voltage outputs of opposite sign, i.e. voltage cancellation when considering the surface average. For practical reasons, due to the space required for the cables connections with the piezoelectric material layers, it would be best to try to minimize the surface coverage for each branch by the piezoelectric material while still ensuring a considerable power output. In the end, the final chosen design for the energy harvester with added proof masses has a minimal partial surface coverage by the piezoelectric material as shown in Figure 5.10.

6

Challenges encountered

During the completion of the project, multiple challenges were encountered related to the creation of the fractal tree geometry, the meshing, the FRFs, the frequency samples and the simulations interruptions. In this chapter all the challenges that have been faced are detailed.

6.1 Risk of branches being too wide

The first type of challenges were related to the fractal tree geometry and most of them are detailed in the *Methodology* section of the report. Yet, another problem which was not previously mentioned was met when generating a bad geometry (see examples in Appendix D), and it occurs when a lower branch (parent branch) is much wider than an upper branch. To prevent that, a numerical security factor was used to ensure that the branches' crossings (intersection points) between one upper and lower branch occur in a way that allows for a minimal upper branch size. Without this safety factor, there would be a risk that an upper branch is engulfed by a lower branch and makes it non existent, see examples (b), (c) and (d) in Figure D.1.

6.2 Distinguishing a bad geometry from a static failure case

In order to enable the GA to distinguish a design with a bad geometry from a design that fails static load for the Phase 0 and Phase 1 optimization setups, the extremely high values given to the objective functions OF1 and OF2 are set differently depending on the case encountered. A bad geometry is associated to objective functions values set to $OF1 = OF2 = 10^{12}$, whereas a failing static load case is associated to objective functions values set to $OF1 = OF2 = 10^{10}$. This is implemented so that the GA will start to converge after fewer iterations by avoiding most importantly a bad geometry. After generating designs with good geometries, the GA will also try to generate designs that do not fail the 10 g-force static load. Hence, after a couple of optimization iterations the GA will generate designs that do not present a static load failure, so that the `MBSD.m` function can proceed with the modal analysis followed by the FRF analysis.

6.3 Mesh absolute repair tolerance for the union of fractal tree geometry sections

A particular fail-safe feature was implemented in terms of mesh absolute repair tolerance, which COMSOL uses to make a geometrical union between the fractal tree shape and the branch sections. Meshing errors which cancel optimization simulations can occur after importing the various branch sections used as piezoelectric material coverage surfaces and the surfaces where proof masses are placed. Sometimes all those different sections and the overall tree geometry cannot be united as a whole geometry.

To prevent the occurrence of meshing errors in COMSOL, first an appropriate starting mesh absolute repair tolerance must be found by trial and error so that at least a mesh can be done even if it has issues and is not coherent. For this project's meshes, an absolute repair tolerance of 6×10^{-5}m proved to be functional. Additional meshing issues can be avoided by implementing an absolute repair tolerance which is modified from the starting value in an iterative manner when the meshing was not successful but has been performed nonetheless. In order to know if a mesh was done successfully, mesh statistics are extracted from COMSOL via the LiveLink for MATLAB function called `mphmeshstats`. The most important mesh information that must be extracted from COMSOL via the variable called *hasproblems* is the notification if meshing problems were encountered.

If the absolute repair tolerance is set too low, COMSOL will not manage to unite the geometries properly, resulting in a mesh that considers all the geometry sections separately, and creating mesh elements that are extremely fine. Figure 6.1 shows an example of a case where this happened.

More mesh information can be extracted, such as the maximal and minimal mesh element volume values through the respective COMSOL variables *maxvolume* and *minvolume.* The mesh can be furthermore controlled and even improved by ensuring that there is no extreme irregularity or disparity between mesh elements size. Yet this was not tested for this project, because the risks of encountering again meshing problems was too considerable.

In addition, the absolute repair tolerance is set to reach a maximal value of 1×10^{-4}m after smaller values have been tested in an iterative way. The problem is that for some designs the mesh will become not coherent if the absolute repair tolerance is too high and it can potentially eliminate some branch sections. Figure 6.2 shows a case of this occurring close to where the mid branches intersect. For stability reasons, the `MBSD.m` code does not increase the absolute repair tolerance more, meaning that if the highest absolute repair tolerance of 1×10^{-4}m is reached, the script skips all computations related to the design at that optimization iteration.

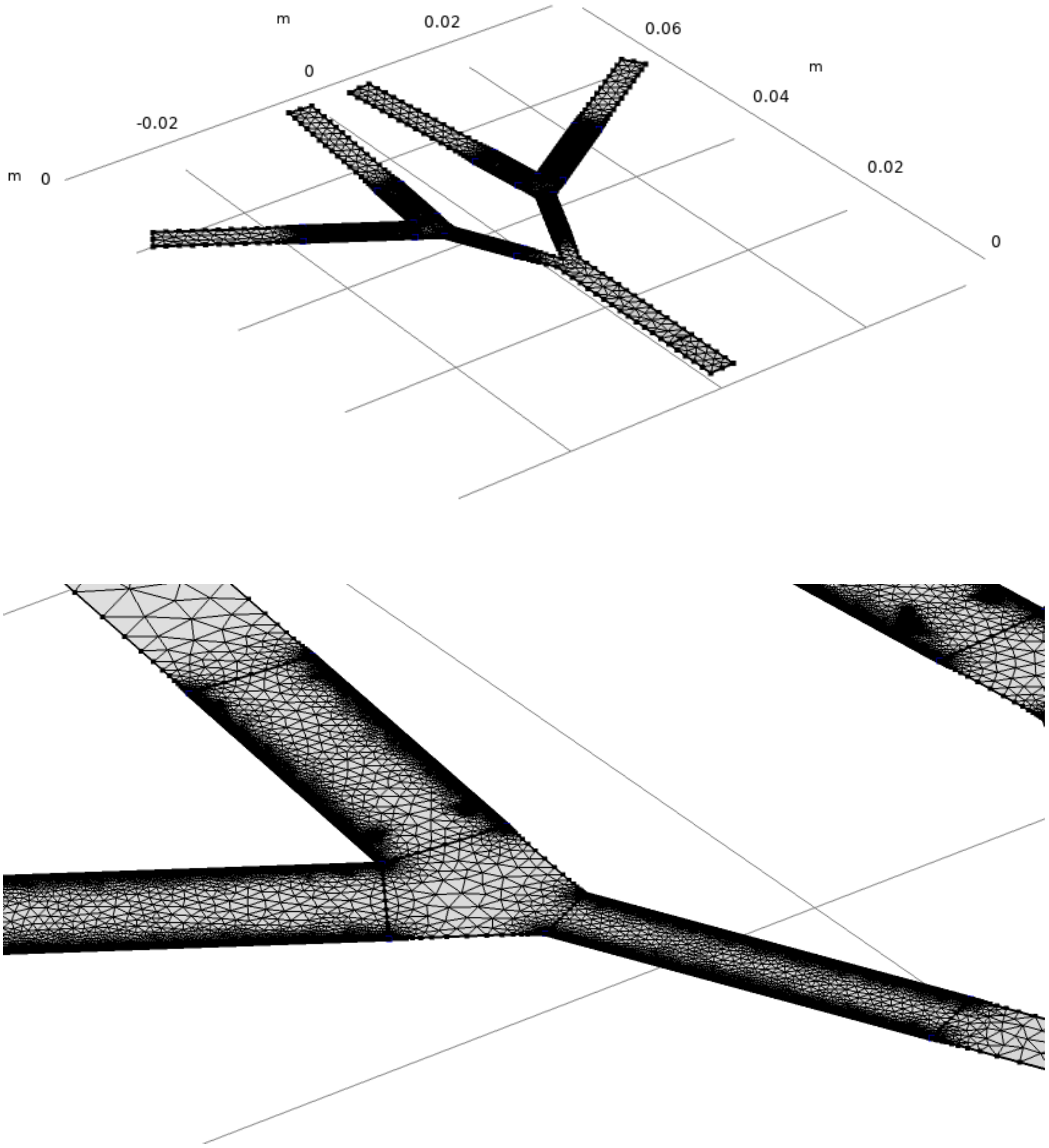


Figure 6.1: Example of a geometry case with the absolute repair tolerance set too low, resulting in a mesh with extremely small elements along some of the edges.

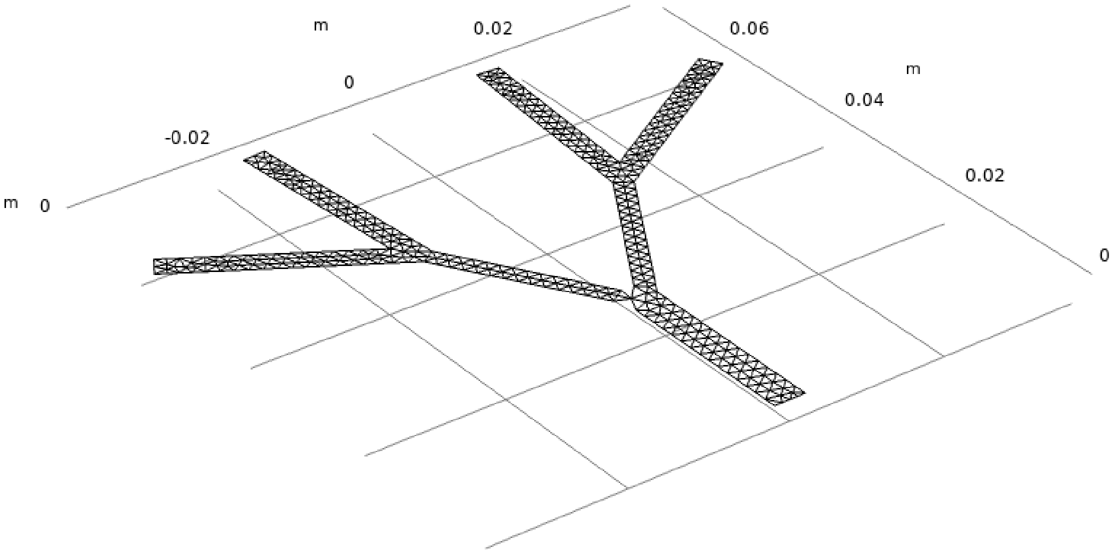


Figure 6.2: Example of a geometry case with the absolute repair tolerance set too high, resulting in a mesh with sections not coherently meshed close to where the mid branches intersect.

6.4 No bandwidth - threshold above all FRF values

With regard to stress FRFs, an issue that could happen is that design variables sets generated by the GA would produce a cumulative longitudinal stress FRF with no bandwidths at all. This would be equivalent to having for a specific design configuration, no voltage output above 100 mV, if there is no FRF longitudinal stress value above a given threshold. Such undesirable design variables are saved as MATLAB .mat files so that they can be checked later after an optimization procedure has been completed. Subsequently, the area under the threshold, meaning the difference between the threshold and FRFs values below it, is also saved as a .mat file.

6.5 Frequency samples too close to eigenfrequencies

Another stress FRF related issue is encountered specifically for FRF values near eigenfrequencies. The FRF values are very steep with a high gradient near the eigenfrequencies due to a low damping. This produces numerically unstable FRF results characterized by sudden drops at eigenfrequencies and for other frequency samples nearby. To counter that issue, the frequency samples located closest to the eigenfrequencies, including the ones being the eigenfreuqencies themselves, are eliminated from the set of frequency samples generated by the geometric progression function (see example in Figure 6.3).

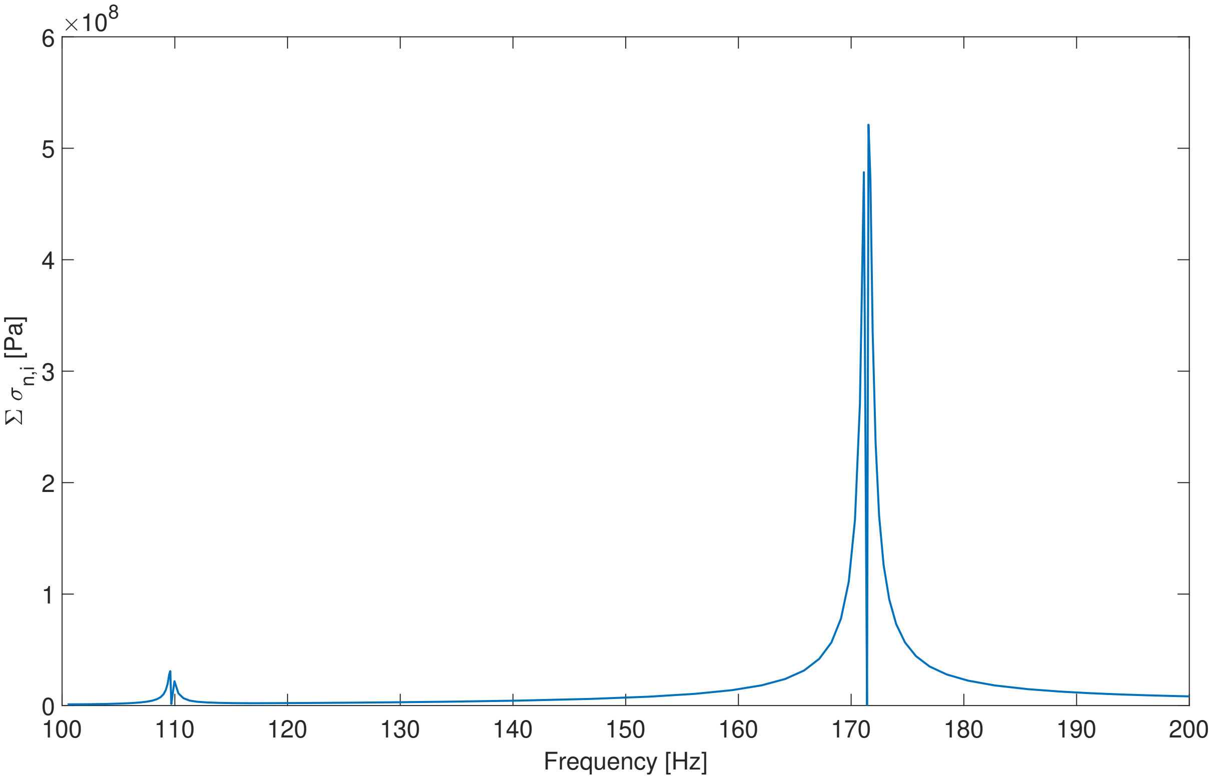


Figure 6.3: Example of total surface averaged longitudinal stress ($\Sigma\sigma_{n,i}$) FRF output, for the design with no proof masses, with frequency samples being too close to the eigenfrequencies.

6.6 Simulations randomly interrupted

Finally, the optimization procedures runs had stability issues causing unpredictable simulation interruptions. The simulations got randomly canceled due to interruptions of connection between the computer and COMSOL's server that is used to run them through LiveLink for MATLAB. Unfortunately that challenge was dealt with by rerunning multiple times certain optimization setups simulations whenever they got canceled unexpectedly.

7
Future work

Future work can be done by improving the FE models accuracy (damping and piezoelectric material coverage), the GA settings (population size and number of generations increased) and the FRFs used for optimization (voltage instead of stress FRF and increasing the number of frequency samples). Such future work prospect is clarified in this chapter.

7.1 Greater population size and greater number of generations - more GA iterations and increased computational time

An important observation about the GA optimization settings, is that one must be careful with the values given to the population size and number of generations. For the Phase 2 optimization setup Pareto front and set results (see Figure 5.6), it can be seen that a small population size and low number of generations is enough for the GA to converge towards a satisfying Pareto front. This is expected since the geometry of the fractal tree design does not vary at each iteration like for Phase 1. Hence, with regard to the Phase 2 optimization setup Pareto fronts, it is observed that increasing the population size and number of generations was not necessary after having arbitrarily chosen a starting population size of 25 with 15 generations. However, for the Phase 1 optimization setup results, a bigger population size and more generations would have confirmed if the final Pareto fronts associated to a population size of 45 with 20 generations are sufficiently accurate (see Figure 5.4). To be able to proceed with the GA simulations associated to a bigger population size and more generations, it would require faster computers or relying on a cluster or local server. This would allow to simulate in a shorter computational time more optimization iterations. This is crucial to avoid having simulations being canceled due to interruptions of connection with COMSOL's server which is used for running LiveLink for MATLAB.

7.2 FE model consideration of piezoelectric material as layered shells

Improvements could also be made in terms of FE model accuracy for the stress FRFs analysis, for instance by considering the addition of layered shell elements to model the piezoelectric material. If thicker, stiffer or more dense piezoelectric materials, other than the PVDF, were to be analysed, then it would be necessary to have layered shell elements added to the FE model used for the stress FRF analysis via COMSOL's Composite

Materials Module. The resulting shift of eigenfrequencies between an FE model with piezoelectric material consideration and without might not be negligible in that case. Implementing layered shells in the FE model used for stress FRFs analysis would obviously require considerably more computational time. Yet, as voltage FRFs results show the assumption of weightless and without stiffness piezoelectric materials holds well for the PVDF piezoelectric material layers that are thin and light. There is no great shift of eigenfrequencies for both Phases 1 and 2 designs, with and without proof masses, between the voltage and stress FRFs (refer to Figures 5.17 and 5.18). Obviously, the current FE model used for stress FRFs analysis is valid for the use of any other thin and light piezoelectric material as well.

7.3 Future consideration of voltage rather than stress FRF for optimization

If computational time was less of an issue, it would be possible to rely for each optimization iteration on the FE model using the COMSOL Composite Materials Module to generate voltage FRFs. This would allow for a more realistic consideration of mass and stiffness due to the piezoelectric material placement on each branch. In those circumstances, it would be also possible to use the voltage threshold of 100 mV in order to determine the voltage FRFs characteristics, such as bandwidths frequency range and FRF area over the threshold. It would not be necessary to convert the 100 mV threshold to a cumulative longitudinal stress threshold, like it is the case now for all stress FRFs analyses carried out.

7.4 Adapting mesh absolute repair tolerance value for a macro-energy harvester to a suitable value for a micro-energy harvester

The two current designs that have been found through the GA optimization procedures are for macro-energy harvesters, considering that they are fitting within an 8 cm × 8 cm area. For future investigations on the design of a micro-energy harvester, another absolute repair tolerance has to be used within the FE models for the union of the fractal tree geometry with the branch sections and to obtain a successful mesh. If the absolute repair tolerance is not appropriate, meshing errors might happen at random optimization iterations.

7.5 Damping

As mentioned in the *Assumptions* section, the damping due to the glue deposited in-between the steel framework and the piezoelectric material is omitted in both the FE models used for stress and voltage FRFs analysis. To obtain the damping of that adhesive interface, experimental data would be required. The consideration of that damping would improve furthermore the FE models accuracy and help for the validation process of the chosen fractal energy harvester design with proof masses.

7.6 Surface averaged stress FRF computation used for optimization

When computing the surface averaged stress FRF for each optimization iteration, the accuracy is not optimal because the average of all nodal values per branch section is computed without considering a nodal weighting factor. The consideration of nodal weighing for each iteration was tested, but it was ignored as an alternative since the simulation time had to be minimized and the results were already approximated with respect to other factors not taken into account. Such factors include damping, as well as mass and stiffness of the piezoelectric material. However for post-processing, a weighted average was used by extracting the surface averaged stress and voltage of each branch section directly through COMSOL's *Average* feature.

Therefore, if necessary, for future work, even if it would require considerably more computational time, it could be relevant to improve the accuracy of FRFs results for each optimization iteration by using COMSOL's *Average* feature to extract surface averaged stresses.

7.7 Additional factors influencing results accuracy

The accuracy of all results exported from FE models can be improved if a mesh convergence study were to be done for each optimization iteration. That has been tested and has added a considerable computational time. On that account, the finest mesh achievable according to the default setting in COMSOL was selected and considered to be appropriate for all optimization simulations, including for the ones carried out for the post-processing procedure.

Besides, the results accuracy is not only affected by the meshing, the surface averaging method, the damping characterization, or by the consideration of the added mass and stiffness due to the piezoelectric material. It is also slightly influenced by the quantity of frequency samples that are used to execute the FRF studies and that are generated by the geometric progression function.

8
Conclusion

The concept of the design for the piezoelectric energy harvester is a symmetric fractal tree design that fulfills a 8×8 cm^2 maximum size requirement. This concept design consists of two beams extending as branches from a parent branch in several levels. The chosen specific symmetric fractal tree concept design has a trunk and two additional levels with seven branches in total.

For this concept design's Phase 1 optimization procedure, length, angle and width for each branch level are varied, with the option to have added identical proof masses placed at the ends of the four top branches. For the Phase 2 optimization procedure, only piezoelectric material placement is varied for each branch.

The energy harvester which is studied is intended to operate up to 200 Hz at 2 g-force for the dynamic analysis and at 10 g-force for the static analysis. It is of interest for the final chosen fractal tree design to provide more eigenfrequencies and bandwidths in a lower usable frequency range. Consequently, the project aim was mostly to optimize FRFs characteristics.

Regarding simulations, computational time is saved by using a geometric progression function to compute concentrated FRF frequency samples close to the eigenfrequencies. By using frequency samples generated by the geometric progression function, numerical precision is increased near eigenfrequencies FRF peaks and computational time is reduced.

For the fractal tree designs with and without added proof masses, the eigenfrequencies were optimized so that they were close enough to each other and produced a sufficiently high cumulative longitudinal stress. Also, a high cumulative longitudinal stress is equivalent to a high electric total voltage output for broader bandwidths (frequency ranges) with a threshold of 100 mV. Even though it is not used for optimization, the maximal displacement at the end of the four top branches of the fractal tree energy harvester is also checked for both static and dynamic FRF analyses.

Later on, a prototype of the chosen final fractal energy harvester design with proof masses will be built. This prototype will be used to validate FRFs from simulations by comparing them to experimental FRFs obtained through Laser Doppler Vibrometer tests. This will be done in the MC2 Laboratory and in collaboration with RISE.

8.1 Undesired scenarios of designs generated by the GA

For the Phases 0 and 1 optimization setup, multiple geometry related design variables were tested by the GA, such as proof mass coverage and weight, and branch dimensions (angles, width and length) with piezoelectric material placement on a full branch surface. During Phases 0 and 1 optimization iterations, a 2D fractal tree geometry was considered bad if it had non-realistic dimensions, undesirable angles between branches, too wide branches, or did not fulfill the maximal size requirement. In that case, high values were attributed to the objective functions when any of those undesired design characteristics were encountered. The same principle of objective functions being attributed high values as a penalty was applied for excluding design variables options that generated an unacceptable high static stress at the trunk root point (0,0,0) of the fractal tree. This was also applied for excluding design variables which had one or more values out of bounds (upper and lower limits). If a design variables option would have generated stress FRFs values that are all under a given threshold, they would have been also penalized in terms of objective functions values, since an optimal design solution would rather maximize the sum of bandwidths and the FRF area above the threshold.

8.2 Final chosen design with added proof masses

The final chosen fractal energy harvester design has added proof masses at the end of the top branches and it has a partial piezoelectric material surface coverage for each branch. It can be seen in Figure 5.10 with dimensions specified in Table 5.7. The proof masses dimensions and weight for the final design are optimal. For the prototype's future manufacturing, multiple proof masses with different materials and dimensions should be considered to find out which one fits best the optimal solution.

8.3 Optimization goal fulfilled

In the end, the goal of the project was fulfilled. It was not intended to achieve the highest accuracy for each optimization iteration, because that would not impact much the Pareto fronts and sets results of optimal design solutions. For example, even though the conversion of the voltage threshold to a stress threshold was estimated by using the mean value of conversion factors FRFs plots, a slight change of the stress threshold value would have not impacted at all the Pareto fronts and sets obtained. The same conclusion applies to the FE models accuracy, the surface averaged stress and voltage FRFs, the mesh refinement, and to the consideration of mass and stiffness added by the piezoelectric material layers when computing stress FRFs.

Bibliography

[1] N. Wu, B. Bao, Q. Wang. “Review on engineering structural designs for efficient piezoelectric energy harvesting to obtain high power output”. *Engineering Structures.* 2021

[2] H. Liang, G. Hao, O. Z. Olszewski. “A review on vibration-based piezoelectric energy harvesting from the aspect of compliant mechanisms”. *Sensors and Actuators: A. Physical.* 2021.

[3] A. Vyas. “Towards an on-chip power supply”. Licentiate thesis. Chalmers University of Technology, 2019.

[4] J. Andersson. “Bandwidth Improvement of a Piezoelectric Micro Energy Harvester”. Master’s thesis. Chalmers University of Technology, 2020.

[5] Seyed Milad Mousavi Bideleh, Viktor Berbyuk. “A Computer Code for Sensitivity Analysis and Multiobjective Optimization: SAMO Tutorial”. Research report. Chalmers University of Technology, 2017.

[6] MathWorks. Genetic Algorithm. https://se.mathworks.com/discovery/genetic-algorithm.html

[7] R. K. Arora. “OPTIMIZATION: Algorithms and Applications”. *Ch. 5: Guided Random Search Methods.* Taylor & Francis Group, LLC. 2015.

[8] MathWorks. Global Optimization Toolbox. https://se.mathworks.com/products/global-optimization.html

[9] MathWorks. Find minimum of function using genetic algorithm. https://se.mathworks.com/help/gads/ga.html

[10] LiveLink™ for MATLAB® User’s Guide. COMSOL, 2018.

[11] S. Ozana, M. Pies, R. Hajovsky. “Using MATLAB and COMSOL Multiphysics for Optimization of the Model of Underground Thermal Processes at Old Mining Dumps”. *Applied Mechanics and Materials, ISSN: 1662-7482, Vols. 548-549, pp 571-578*, 2014.

[12] H. Li, Q. Zhang. “Multiobjective Optimization Problems With Complicated Pareto Sets, MOEA/D and NSGA-II”. *IEEE Transactions on Evolutionary Computation, VOL. 13, NO. 2*, APRIL 2009.

[13] P. Ngatchou, A. Zarei, M. A. El-Sharkawi. “Pareto Multi Objective Optimization”. University of Washington, Seattle, WA, 2005.

[14] V. Berbyuk. “TME146 Structural Dynamics Control, Lecture 9: Multi-Objective Optimization”. Presentation. Chalmers University of Technology, 2020.

[15] K. Deb, A. Pratap, S. Agarwal, T. Meyarivan. “A Fast and Elitist Multiobjective Genetic Algorithm: NSGA-II”. *IEEE Transactions on Evolutionary Computation, VOL. 6, NO. 2*, APRIL 2002.

[16] Comsol. Rayleigh Damping.
https://doc.comsol.com/5.5/doc/com.comsol.help.sme/sme_ug_modeling.05.083.html

A

Appendix: Optimization Input parameters

For each optimization procedure iteration, the `SAMO.m` MATLAB code does an update of the design parameters (or design variables) from the Excel file `InputParams.xls`. Those updated design parameters are used as inputs for the `MBSD.m` function. Afterwards, the design parameters from the current optimization iteration are used in the `MBSD.m` function to compute the objective functions which are intended to be minimized.

It is undesired that any of the design parameters at any iteration is out of the bounds, fixed by the user initially in the Excel input file `InputParams.xls`, or that other constraints too are violated (feasible geometry, FRFs values above threshold, static stress limit respected). If any of such constraints happens to be violated at an optimization procedure iteration, then, as penalty high values are given to the vector of objective functions. That is done in order to ensure that the Pareto optimized results are rightfully selected by the GA without ever having bad design parameters results selected.

A.1 Input parameters Excel file

The optimization procedure input and the Elitist Genetic Algorithm (GA) settings for the Global Sensitivity Analysis (GSA) and the multi-objective optimization are set in the Excel file called `InputParams.xls`.

The objective functions sensitivity analysis has three parameters which are set in `InputParams.xls` and those are: the covariance (COV), the number of points (No. Pts) and the distribution type (Dist) of the data with the given number of points and the mean value being provided. The two distribution types which can be applied for the objective functions sensitivity analysis are: log-normal ("l" or "L") and normal ("n" or "N") [5].

As for the multi-objective functions parameters, there are some which are associated to the bounds of the design parameters and some which are only related to the Genetic Algorithm settings. That being said, the lower bounds and upper bounds parameters are associated to the design variables permissible values to be considered for computing the objective functions. If a Pareto set with certain objective functions values has any design parameter out of bounds, then those objective functions are penalized (artificially increased numerically) by the SAMO code. On the other hand, the Genetic Algorithm parameters are: the population size, the number (No) of generations, the Elite count, and the Pareto fraction.

B

Appendix: Sensitivity Analysis

The Global Sensitivity Analysis (GSA) is performed by SAMO after it reads the design variables data provided by the `InputParams.xls` and also after getting the objective functions provided by the `MBSD.m` code. The GSA results are displayed as total sensitivity indices for each one of the different objective functions and also for each one of the design parameters (or design variables) [5].

The results are quantified as a color bar table which indicates the level of sensitivity. In this color bar table, low values (blue tones) are associated to a low sensitivity with respect to a certain design parameter and objective function, whereas high values (red tones) are associated to a high sensitivity. In the example from Figure B.1 below, the SA results are taken from the Phase 0 optimization setup. There are three objective functions, i.e. OF1, OF2 and OFC, numbered from 1 to 3 respectively for the SA which is carried out in this example. The OF1 is the minimization of the 1^{st} eigenfrequency, the OF2 the minimization of the 5^{th} eigenfrequency, and the OFC represents what is defined as an objective function condition. That objective function is related to multiple conditions needed to be fulfilled by the fractal tree design. For this GSA example, the design parameters are the geometry variables of the fractal tree, i.e. L, L_2, L_3, α, α_3, w, w_2 and w_3, which are respectively numbered from 1 to 8 in Figure B.1. From this Figure it is clear that OF1 and OF2 have a higher sensitivity with respect to L, L_2 and L_3.

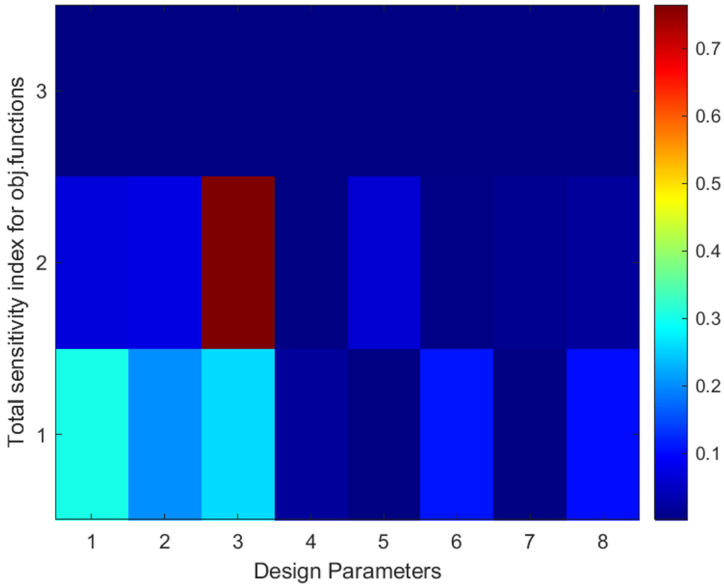


Figure B.1: Example of SA plot.

C

Appendix: Tree geometry intersection points

The fintersect.m MATLAB function, just like fintersect_modified.m, requires for each curve with equation $y = ax + b$ the following inputs: the angles in degrees defining the curve slope, the coordinates x and y of one point which is part of each curve, the slope factor a and the slope constant b. In the circumstances that a curve is actually an abscissa value ($x = constant$) defined by a curve angle of 90 degrees value (or infinite slope factor a), then the intersection occurs at the point where the other curve reaches that abscissa value. Otherwise the intersection point ($x_{intersection}$, $y_{intersection}$) is computed by using the slope factors a_1 and a_2 and the curve constants b_1 and b_2 in the following way :

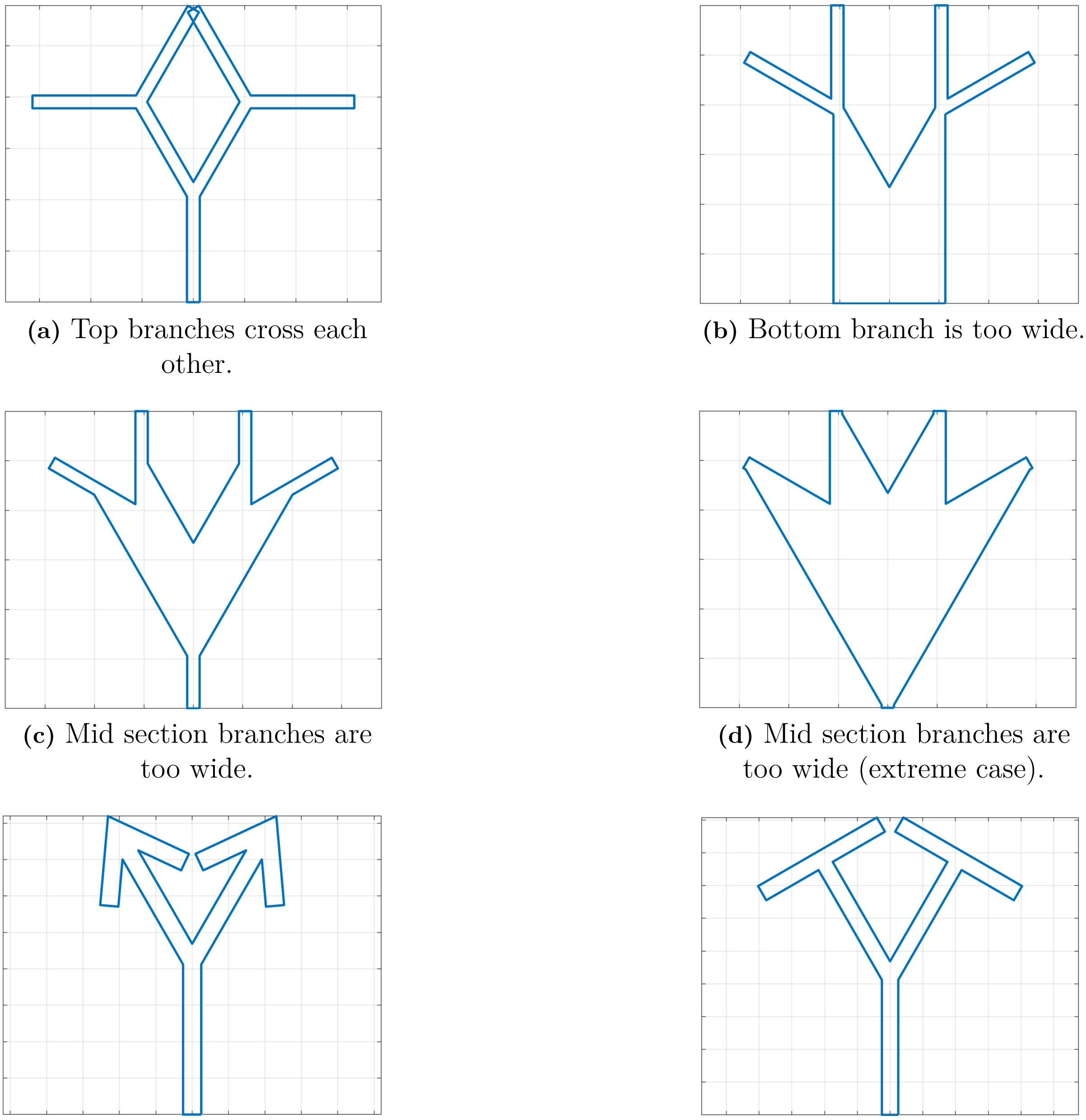
$$\begin{aligned} x_{intersection} &= \frac{b_2 - b_1}{a_2 - a_1} \\ y_{intersection} &= a_1 \cdot x_{intersection} + b_1 \\ &= a_2 \cdot x_{intersection} + b_2 \end{aligned}$$

Another MATLAB function which is used within both functions Tree_code_function.m and Tree_code_function_modified.m, is called fmidpoint.m. This function calculates by averaging the coordinates x and y of a midpoint ($x_{midpoint}$, $y_{midpoint}$) between two other points, with coordinates (x_1, y_1) and (x_2, y_2) respectively :

$$\begin{aligned} x_{midpoint} &= \frac{x_1 + x_2}{2} \\ y_{midpoint} &= \frac{y_1 + y_2}{2} \end{aligned}$$

D

Appendix: Examples of unwanted tree geometries



(a) Top branches cross each other.

(b) Bottom branch is too wide.

(c) Mid section branches are too wide.

(d) Mid section branches are too wide (extreme case).

(e) Angle $\alpha_3 > 180°$.

(f) Angle $\alpha_3 = 180°$

Figure D.1: A few examples of unwanted geometries.

E

Appendix: Objective function condition - `OFC`

The objective function condition checks first if geometrical conditions are respected and then if the static load does not lead to a static failure. If the first case is encountered, the eigenfrequency study and the FRF simulation are skipped, and if the second case is encountered afterwards the design configuration is discarded.

E.1 Geometrical conditions for Phases 0 and 1

When at each optimization iteration a fractal tree geometry is generated, it is checked to see if it will meet given geometrical conditions. If it fails to do so, the optimization procedure needs to discard that geometry. A specific objective function is therefore set to indicate that. It is called `OFC` (objective function condition) in the MATLAB codes, and it is set to value 0 if the design geometry fulfills all the geometrical conditions or to value 1 if the geometry instead fails to do so.

E.2 Static load for Phases 0 and 1

The static load case is the first study to be analysed if the fractal tree geometry fulfills the geometrical conditions. If a fractal design fails the static load, the same objective function condition `OFC` is given a value 1, to signal that the generated geometry might break. If a fractal design sustains the static load without failure, the value of `OFC` remains 0 and the optimization procedure iteration continues to do the eigenfrequency study and then the FRF study.

F

Objective functions to be minimized - OF1 and OF2

F.1 Phase 0

In the first sets of analysis for the Phase 0 optimization setup, the objective functions that are chosen to be minimized are the 1^{st} eigenfrequency and the 5^{th} eigenfrequency for a tree geometry without proof masses considered. Those eigenfrequencies are delimiting the frequency range of analysis from 0 Hz to 200 Hz, because the 5^{th} eigenfrequency is for most geometries being tested by the GA, the largest eigenfrequency below 200 Hz. These two objective functions are defined as `OF1` and `OF2` in the optimisation codes respectively. The reason for choosing these are that they are eigenmodes which, according to some previous analysis prior to the Phase 0, seem to have eigenshapes that are more beneficial in bending the branches of the fractal energy harvester design. An example of a previous modal analysis of the first five eigenmodes can be seen in Appendix M. Such eigenshapes, which are characterized mainly by a bending shape rather than a torsional or twisting shape, are more desirable because they generate higher stresses and thus also higher voltage outputs. The eigenmodes in between, i.e. second, third and fourth, do instead have more various twisting shapes along the branches. It is assumed that by minimizing the 1^{st} and 5^{th} eigenfrequencies, the remaining eigenfrequencies in between will follow along, but not actively be minimized.

F.2 Relation between OF1 and OF2 with OFC

For both Phase 0 and Phase 1, the geometric shapes of the fractal energy harvester that are given a value of 1 for the `OFC`, need also values for `OF1` and `OF2`. Since they are unwanted and need to be penalized, they are given in the codes an artificial high value of 10^{12}. A similar approach is done for the objective functions `OF1` and `OF2` when the geometry fails the static load. Under these circumstances, the fractal energy harvester configuration is penalized with `OF1` and `OF2` being attributed an artificial high value of 10^{10}.

F.3 Phase 1 and Phase 2

In the later part, for both the Phase 1 and 2 optimization setups, two other objective functions are considered which are two FRF characteristics. The first objective function intended to be maximized is the sum of all stress FRF frequency bandwidths, and the other is stress FRF areas above the threshold. For the Phase 2 optimization setup, a new objective function is introduced which quantifies the power generation per branch through a multiplication of piezoelectric material area coverage per branch with the corresponding surface averaged stress FRF.

For the Phase 1 optimization setup, if the fractal tree geometry at any optimization iteration is bad, then the objective functions are assigned very high values. The objective function condition (OFC) in that case is assigned a value of 1 instead of 0. If at any optimization procedure the fractal tree static von Mises stress for the trunk root point (0,0,0) is over the stress limit (yield stress divided by security factor), the objective functions are assigned high values too, yet less high than if there is a bad geometry occurring. Once again, in the case of a non permissible maximal static von Mises stress too, the objective function condition (OFC) is assigned a value of 1 instead of 0.

In summary, depending on the undesired case scenario that happens the objective functions are assigned high values as shown in Table F.1.

Table F.1: Phase 1 optimization objective functions at an iteration with values that are arbitrarily set when a bad geometry occurs or when static stress at point (0,0,0) is too high.

Undesired case scenario	**OF$_1$**	**OF$_2$**
Bad geometry	10^{12}	10^{12}
Static stress (σ_{vM}^{max}) too high	10^{10}	10^{10}

G

Appendix: Phase 1 optimization setup Static analysis data computed in COMSOL and imported in MATLAB

For the Phases 0 and 1 optimization setup, in the `run_comsol_static_function.m` code, the static stress and displacements results are exported from COMSOL as .txt files. They are converted to .mat files for MATLAB by running the code `txt2mat_static.m`. The `txt2mat_static.m` MATLAB script consecutively uses the MATLAB function `readMultiphysicsTXT_function.m` for each filename with extension .txt as an input and then converts it to .mat.

Thus, any relevant COMSOL result can be exported as a .txt file and then converted to a .mat file by using the MATLAB function called `readMultiphysicsTXT_function.m` which requires a filename as an input.

For the analysis of the von Mises stress at a reference frequency prior to the Phase 2 optimization setup, and for the extraction of branch surface sections coordinates related to von Mises FRF stress values, it is necessary to use instead the function `readMultiphysicsTXT_function_modified.m`.

H

Appendix: Eigenfrequencies computed in COMSOL and imported in MATLAB

H.1 Eigenfrequencies computed in COMSOL, exported as .txt files and converted to .mat files

In the Phase 1 optimzation setup, for each optimization procedure iteration, the eigenfrequency analysis is performed in `MBSD.m` via the MATLAB function `run_comsol_eig_frf_study_function.m`, which runs the eigenfrequency analysis that is set up in COMSOL. The eigenfrequencies are exported from the COMSOL simulation as a .txt file and then directly imported in the MATLAB workspace as a .mat file by running the function `readMultiphysicsTXT_function.m`. For the Phase 2 optimization setup and the procedure prior to it, the eigenfrequency studies are conducted respectively by running `run_comsol_eig_frf_study_function_PIEZO.m` and `run_comsol_eig_frf_study_function_modified.m`.

H.2 Eigenfrequencies loaded in MATLAB

In order for any of the eigenfrequencies to be set as an objective function for minimization for the Phase 0 optimization setup, all eigenfrequencies need to be imported again as .mat files in MATLAB through the `MBSD.m` code. To import the eigenfrequencies as .mat files in MATLAB, the function `Loading_Comsol_mat_files_MBSD_eig_study_only.m` is used in the `MBSD.m` to load the COMSOL eigenfrequencies .mat results files. After they have been already imported in MATLAB's workspace, any pair of eigenfrequencies can be set as objective functions intended to be minimized.

Similarly, for the localization of maximal von Mises stress per branch prior to the official Phase 2 optimization setup, the function `Loading_Comsol_mat_files_MBSD_eig_study_only_modified.m` is used in `MBSD_modified.m` to load the eigenfrequencies and then make them outputs for that function. Afterwards, in `MBSD_modified.m`, those eigenfrequencies are used as inputs for the longitudinal stress FRF analysis function `Find_bandwidth_Matrix_bandwidths_areas_STRESS_modified`.

I

Appendix: Geometric progression function for FRF frequency samples

The purpose of `geometric_prog_function.m` is to generate smaller frequency steps close to to the eigenfrequencies, and larger steps away from them. This is done with the help of a geometric progression or geometric series.

If a is the frequency step or distance between the first two frequency samples, and r is the size ratio of the next step, the size of the n^{th} step can be calculated as $a_n = ar^{n-1}$. The total sum of all frequency steps within a specific frequency range can be calculated as $L = a(1-r^n)/(1-r)$. By taking half the distance between two eigenfrequencies and defining this as L, this results in the first step size away from the first eigenfrequency being $a = L(1-r)/(1-r^n)$. The step size to the next frequency is then $a_n = ar^{n-1}$. By first defining all frequency steps to the midpoint between two eigenfrequencies, the frequency steps from the midpoint to the next eigenfrequency is then simply mirrored or flipped, so that the step size decreases when approaching the next eigenfrequency.

This process of frequency steps defined as a geometric progression between two specific eigenfrequencies, is repeated as many times as there are eigenfrequencies in the frequency range of interest (FRF lower frequency limit up to FRF upper frequency limit) which is used to compute the FRF analysis. However for the first and last eigenfrequencies present in the FRF frequency range of interest, the half-distance L is no more the distance between two eigenfrequencies, but rather the distance between those eigenfrequencies and their corresponding frequency limit. Finally, with all frequency step sizes given, the actual frequency samples are calculated and imported to COMSOL for the FRF analysis. In Figure I.1 an example is shown of frequency samples that are generated by the geometric progression.

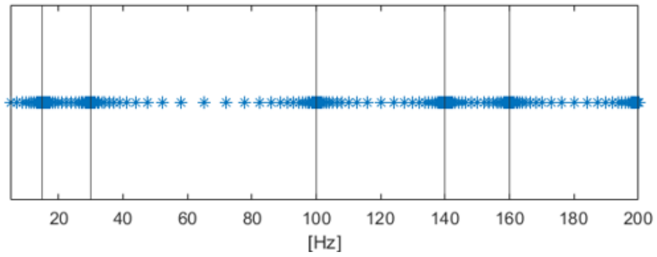


Figure I.1: Example of how frequency samples are distributed. Blue asterisks show frequency samples, black vertical lines are eigenfrequencies.

J

Appendix: FRF analysis in MATLAB via FRF data computed in COMSOL and imported in MATLAB

For the Phase 1 optimization setup in the `run_comsol_eig_frf_study_function.m` code, the FRF for stress and optionally for displacements results, are exported from COMSOL as .txt files and converted to MATLAB .mat files. This is done by running the code `txt2mat.m` which converts all relevant FRF results .txt files to .mat files. Just like `txt2mat_static.m`, the `txt2mat.m` MATLAB script consecutively uses the MATLAB function `readMultiphysicsTXT_function.m` for each filename with extension .txt as an input and then converts it to .mat.

Alternatively, for the computation of local maximal von Mises stress per branch before proceeding with the Phase 2 optimization setup, the code `run_comsol_eig_frf_study_function_code_modified.m` uses instead the script `txt2mat_modified.m`. Compared to the original `txt2mat.m`, the script `txt2mat_modified.m` converts additionally voltage .txt files FRFs to .mat files. That function also occasionally relies on the function `readMultiphysicsTXT_function_modified.m` to extract coordinates of branch section FE model nodes that are used to pinpoint the location of maximal von Mises stress per branch for a given frequency. That reference frequency can be unique to a branch or common among some branches.

When proceeding afterwards with the Phase 2 optimization setup, `run_comsol_eig_frf_study_function_code_PIEZO.m` is used instead and it relies also on `txt2mat.m` like in the case of the Phase 1 optimization setup to convert to .mat the .txt results files which are exported from COMSOL.

K

Appendix: FRF data characteristics computed to be used as optimization objective functions

The MATLAB function `bandWidth_function.m` is used within the FRF function `Find_bandwidth_Matrix_bandwidths_areas_STRESS.m`, to compute for each single bandwidth the indices of the lower and upper frequency limits. Those two indices are used in `Find_bandwidth_Matrix_bandwidths_areas_STRESS.m` to compute the actual lower and upper frequency limits, the difference between the two frequency limits and their indices. By knowing the index range (lower and upper index) for a single bandwidth, the bandwidth FRF area above a given threshold can be approximated using the MATLAB `trapz` function, which requires as inputs the bandwidth frequencies and the FRF corresponding values in that frequency range. A typical FRF example with the presence of bandwidths and area over a threshold is illustrated in Figure K.1.

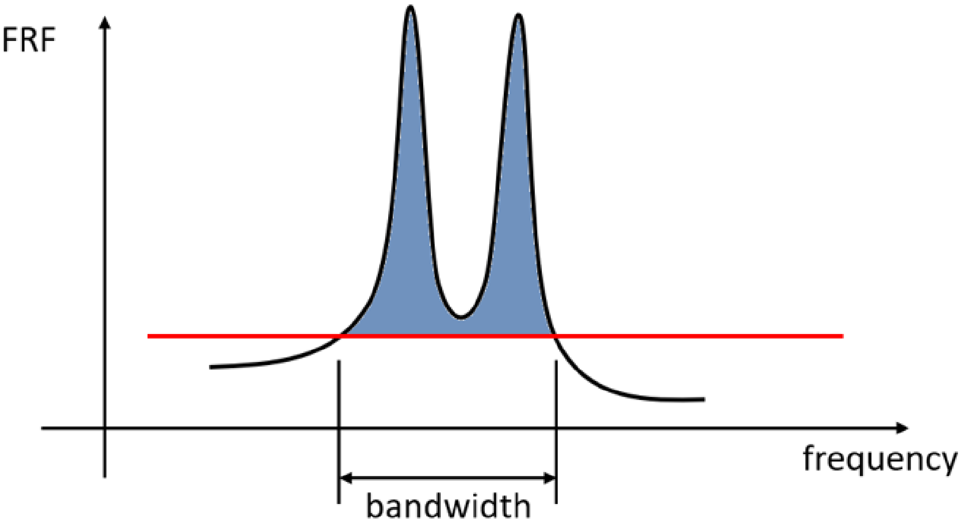


Figure K.1: Frequency Response Function (FRF) bandwidth and threshold value (red line). Area above threshold is in blue.

The following problem can arise while running `Loading_Comsol_mat_files_MBSD_FRF_study_only.m` for the Phase 1 optimization setup or `Loading_Comsol_mat_files_MBSD_FRF_study_only_PIEZO.m` for the Phase 2 optimization setup at a specific optimization iteration. A resulting FRF for stress could generate the presence of no bandwidth, or in other words no area over a given threshold, but rather an area only under that threshold. It is important to ensure that these undesired design variables are not selected by the GA optimization process with their associated objective functions and not considered part of the Pareto set of optimal design solutions (optimal minimized objective functions). In such circumstances, the FRF related objective functions must be penalized. The area under the threshold and the associated undesired design variables are then stored in separate .mat files. In order to penalize such FRF related objective functions, they are attributed a very high numerical value by replacing an objective function of an FRF area over a threshold by the total area under a threshold. This area under an FRF threshold is computed by the difference between the threshold value and the FRF values and then approximated by using the MATLAB function `trapz.m`. Similarly, for an objective function considering normally a bandwidth, that objective function is replaced in these circumstances by the inverse of a small numerical value, such as 1/0.000000000001. For example, this objective function high value represents an undesired nonexistent bandwidth which is associated with a bad design.

All this considers the same principle which is applied in `MBSD.m` when other design conditions are undesirable, such as not respecting geometrical conditions and having an unacceptably high static stress. The same idea is applied in `SAMO.m` when some design variables are selected by the GA and any of them is out of bounds, meaning smaller than the lower bound or greater than the upper bound.

Moreover, a specific bandwidth frequency range is also differently set if only its lower frequency bound is precisely determined in the frequency range of FRF analysis from 0 to 200 Hz. In that case, the bandwidth upper frequency bound is not exactly computed because it would be above the frequency limit of 200 Hz, and consequently it is set to 200 Hz automatically. This implies that the Genetic Algorithm would favour designs with larger stress FRFs bandwidths computed, meaning designs with bandwidths upper and lower frequency bounds which are located as much as possible within the frequency range of interest from 0 to 200 Hz.

L
Appendix: COMSOL

L.1 COMSOL LiveLink for MATLAB

COMSOL LiveLink for MATLAB is a feature of COMSOL Multiphysics that enables COMSOL simulations to be run through MATLAB. With this functionality, the input parameters or settings can be modified and the COMSOL model can be solved using MATLAB scripts.

Useful command that can be used to work with a COMSOL model in MATLAB are the `mphload` and `mphsave`. `mphload` lets the user load the COMSOL model given as a .mph file into the MATLAB workspace. The user might need to find a certain command that will modify or set up the FE model. By using `mphsave` in MATLAB the model can be saved either as an .mph file for COMSOL or as an .m file for MATLAB. Saving it as an .m file will list all the settings of the model in chronological order, including all historical modifications, as a MATLAB code. This implies that the last settings that are being set within the COMSOL model will appear at the last rows of the MATLAB code when saved as an .m file. This is a very useful way to find MATLAB commands that can be used in the optimization MATLAB scripts needed for the optimization procedures.

L.2 Importing in COMSOL the Fractal tree geometry generated in MATLAB

The geometry of the fractal energy harvester is imported in the COMSOL model from the .txt file that is generated by `Tree_code_function.m` for the Phases 0 and 1 optimization setup and by `Tree_code_function_modified.m` for the Phase 2 optimization setup. Those .txt files contain point coordinates that are then used to form the fractal energy harvester shape via the *Polygon* feature in COMSOL. The idea is to import different geometry parts for the Phase 2 optimization setup in a similar way, but by defining additionally individual sections of the geometry. For the Phase 2 optimization setup, the branch sections are varied for each optimization procedure iteration with proof mass sections kept as constant when the design has proof masses placed at the end of the top branches. Then, the resulting geometry considering a varying piezoelectric material coverage area per iteration is saved also as a .txt file by the `Tree_code_function_modified.m` function.

L.3 COMSOL FE models

In COMSOL, the fractal tree geometry is 2D with no solid elements and with additional layered shell elements used only for the voltage computation COMSOL file. The thickness of shell elements which form the steel tree structure is set at 0.2 mm. This applies for both COMSOL FE models, the one used for stress FRF analysis and the one used for voltage FRF analysis. The FE model used only for voltage FRF analysis, and built via the COMSOL Composite Material Module, includes the 0.028 mm thickness of layered shell elements which form the piezoelectric material deposited on top of each branch. Also, the materials, that are obtained from COMSOL's material library, are set to be structural steel for the fractal tree structure and PVDF for the piezoelectric material layers.

L.3.1 COMSOL Layered shell FE model with piezoelectric material voltage coupling

The first step in the FE model used for voltage FRF analysis is to define a *Layered material*. The structural steel is defined as the bottom layer while the PVDF piezoelectric material constitutes the top layer. There are two different types of material sections for the fractal tree geometry. One section type has only structural steel, while the other has additionally PVDF placed on top and is defined as a new layered material.

By using the *Electric Currents in Layered Shells* COMSOL setting, the piezoelectric material is set to be connected to *Ground* through the interface of bottom and top layer.

As for additional boundary conditions applicable only to this particular FE model, a *Continuity* boundary condition is set between interfaces (edges) where there is a transition between the single layer steel material and the defined multi-layer sections with added PVDF.

The other boundary condition applied to this FE model is an *Electric Insulation* on the surrounding edges of the multi-layered sections. This ensures that voltage generation can only be transmitted through the contact interfaces between piezoelectric material (PVDF) and the framework (steel).

L.4 COMSOL Static analysis

The fractal energy harvester must be able to sustain sudden impacts, i.e. if it is dropped to the ground it should not break. The corresponding load case for this scenario is a 10 g-force static load.

To simulate this, the fractal energy harvester model is clamped at the bottom edge of the main trunk branch (see Figure 2.3). This is done by fixing all 6 degrees of freedom along the bottom edge of the trunk. A gravity condition is set to act in negative z direction with a magnitude of $10g$.

When this load case is simulated it is assumed that the maximum stress occurs at the root point (0,0,0), i.e. the middle point along the clamped edge. For both static and dynamic analysis, this point (0,0,0) is created in COMSOL in order to ensure that the mesh is enforced to have that point as a node. This ensures also that the value at point (0,0,0) is calculated directly without being interpolated.

When solving this study in COMSOL, the von Misses stress at this point is set to be exported to a .txt file.

L.5 COMSOL Eigenfrequency analysis

Another study is added in the COMSOL model which performs an eigenfrequency (modal) analysis. The same type of boundary condition is set along the bottom edge of the main trunk (see Figure 2.3) as used in the static analysis (clamped condition). The eigenfrequency analysis is set to solve for the first 10 eigenfrequencies.

L.6 COMSOL FRF analysis

A final study that is added for the Phase 1 optimization setup after the static and modal (eigenfrequency) analysis are completed, is a longitudinal stress frequency response function (FRF) for each branch. On the other side, in the Phase 2 optimization setup, there is only a longitudinal stress frequency response function (FRF) which is being calculated. The range of frequencies that was initially simulated started from 5 Hz and continues to 200 Hz with steps of 1 Hz. However, with the implementation of a geometric progression function, for each optimization iteration, frequency samples are generated with a concentration around eigenfrequencies. This ensures that the FRF are overall more accurate by computing more precisely the peak values at eigenfrequencies (high gradient region), while avoiding using many frequency samples where the FRF does not vary much (low gradient region). For each optimization iteration, the frequency samples generated by the geometric progression function are saved as a .txt file that is afterwards imported in COMSOL and used for the dynamic analysis. For the computation of voltage FRFs used for getting the conversion factor prior to Phase 2, these FRFs are extracted and computed only once, not at each optimization iteration.

The load case for the longitudinal stress FRF study for both Phases 1 and 2 optimization setups, is a 2 g-force which is set to be maximum at 200 Hz. The boundary condition for this case is set to be a prescribed displacement in the out of plane direction along the bottom edge of the trunk (see Figure 2.3). The amplitude of this displacement corresponds to a cyclic 2 g-force at 200 Hz (see *Theory* section 2.4).

The longitudinal stress FRF per branch is also used for the Phase 2 optimization setup. However, before the Phase 2 simulations are carried out, the von Mises FRF stress is computed in order to find the location for each branch where stress is maximal at a reference frequency. As a reminder, the von Mises stress is also computed for the Phase 1 optimization setup in a static analysis, when a 10 g-force is applied while the bottom edge of the main trunk branch is clamped.

L.7 Exporting COMSOL results and importing them in MATLAB

There are two COMSOL files being used for analysis, one is solely responsible for computing the voltage FRF for each branch, whereas the other one computes static stress and dynamic stress FRF for each branch.

For the COMSOL file used for extracting static stress and dynamic stress FRF, in order to export data results, datasets need to be created. The first type of datasets are *Cut Point 3D*, which are different for each geometry point of interest and for each specific study, either static or dynamic FRF analysis. These points are the eight free end corners of the four top branches and the middle point along the bottom edge of the trunk at coordinates (0,0,0). On a side note, the free end corners are updated at each optimization procedure iteration to take into account the updated fractal energy geometry. Yet, the middle point along the bottom edge of the trunk remains unchanged.

Additionally, multiple *Surface* datasets are created for both COMSOL files. One type of *Surface* dataset computes the branch surface FRF stress values, and another type computes directly in COMSOL the surface averaged stress FRF values. As for *Surface* datasets used for voltage FRF analysis, they compute directly the surface averaged values. In summary, a different *Surface* dataset must be created for each study and for each branch section surface from which either stress or voltage FRF results are exported (nodal values). This results in a value for each frequency sample and node of a specified surface for the FRF analysis, and for individual nodal values for the static analysis. The purpose of exporting for a COMSOL file the surface stress or voltage for all branch sections is to make it an absolute value of the surface average and then add all of them together. One can also consider the maximum stress for each branch section surface for the FRF analysis.

Now for the second COMSOL file used for voltage FRF computation, one desires to extract each branch surface averaged voltage FRF directly from COMSOL through the *Average* feature. This generates more accurate values but it requires more computational time. The surface averaged voltage FRFs values are then used to compute a conversion factor FRF of voltage over stress. The conversion factor FRF is afterwards transformed into a scalar conversion factor by taking the average of the FRF values. This scalar conversion factor is finally used to convert the 100 mV voltage threshold into a stress threshold. This stress threshold is intended to be used for the maximization of FRF characteristics for both optimization Phases. There is a MATLAB function called `conversion_factor_stress2volt_function.m` which is used in `MBSD_modified.m` to compute the conversion factor. The conversion factor computed by `conversion_factor_stress2volt_function.m` is used by two MATLAB functions, which are `Find_bandwidth_Matrix_bandwidths_areas_STRESS.m` and `Find_bandwidth_Matrix_bandwidths_areas_STRESS_modified.m`.

M

First 5 eigenmodes for an example of a fractal energy harvester geometry

Figures M.1 to M.5 show examples of typical shapes of 1^{st} to 5^{th} eigenmodes of a fractal energy harvester geometry.

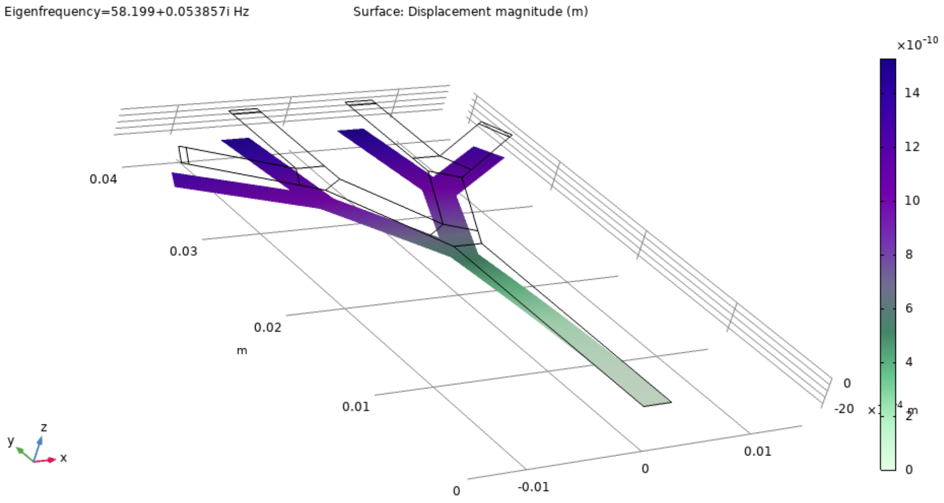


Figure M.1: Example of 1^{st} eigenmode shape.

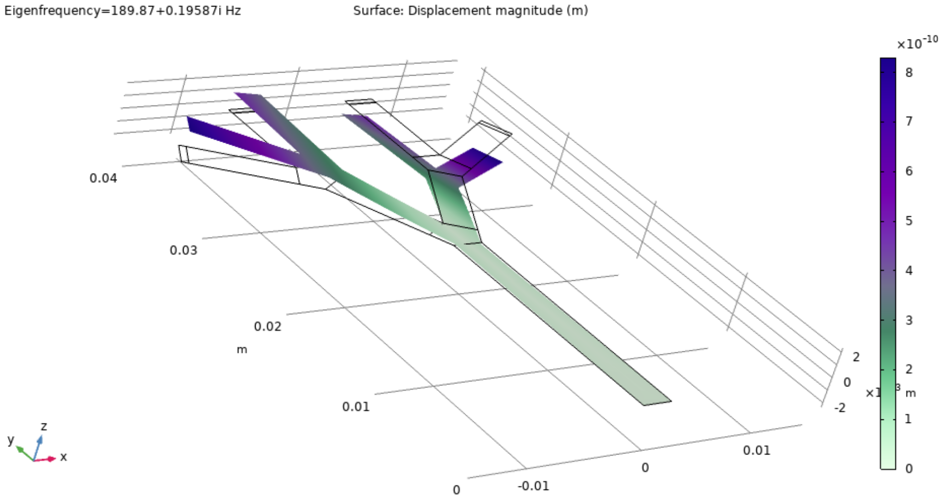


Figure M.2: Example of 2^{nd} eigenmode shape.

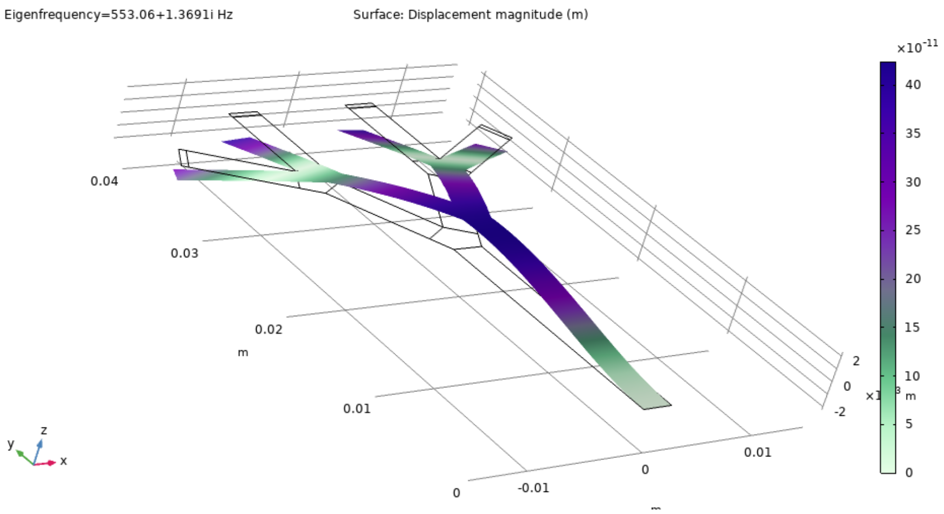

Eigenfrequency=553.06+1.3691i Hz
Surface: Displacement magnitude (m)
×10-11
40
35
30
25
20
15
10
5
0
0.04
0.03
0.02
m
0.01
0
-0.01
0
0.01
2
0
-2
m
y
z
x

Figure M.3: Example of 3^{rd} eigenmode shape.

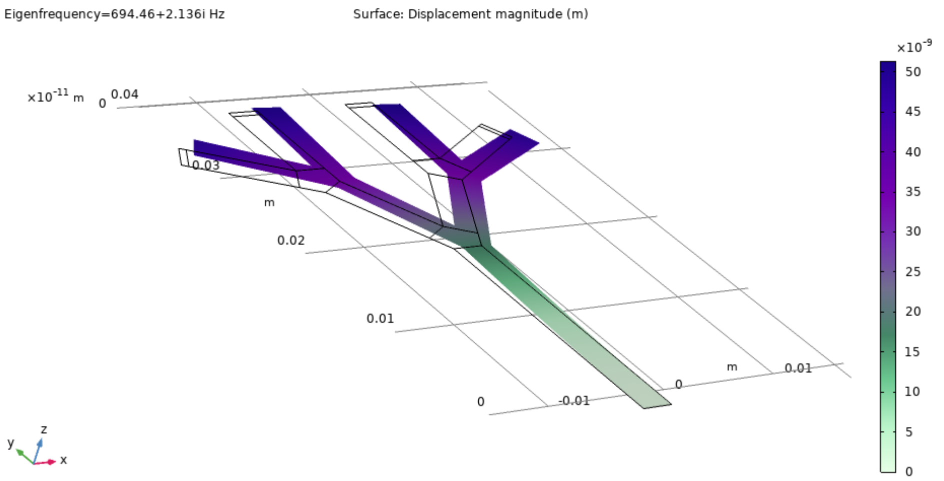

Eigenfrequency=694.46+2.136i Hz
Surface: Displacement magnitude (m)
×10-9
50
45
40
35
30
25
20
15
10
5
0
×10-11 m
0
0.04
0.03
m
0.02
0.01
0
-0.01
0
m
0.01
y
z
x

Figure M.4: Example of 4^{th} eigenmode shape.

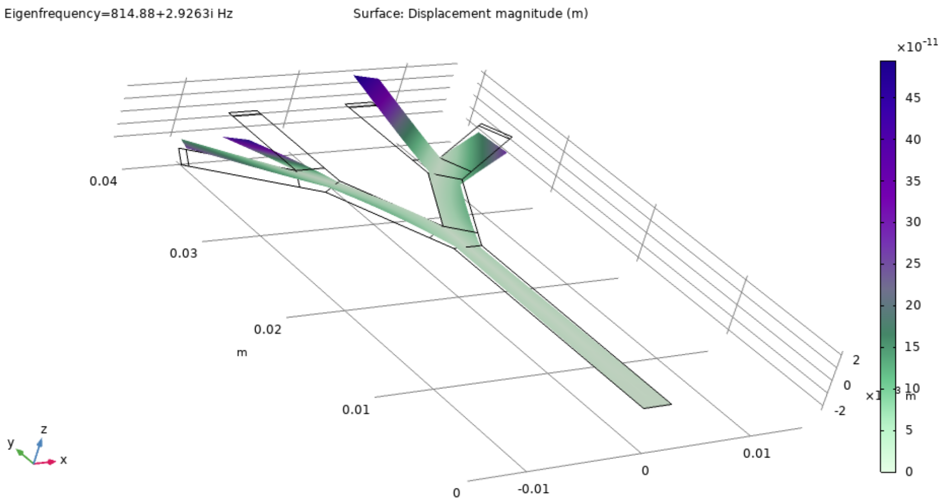

Eigenfrequency=814.88+2.9263i Hz
Surface: Displacement magnitude (m)
×10-11
45
40
35
30
25
20
15
10
5
0
0.04
0.03
0.02
m
0.01
0
-0.01
0
0.01
2
0
-2
m
y
z
x

Figure M.5: Example of 5^{th} eigenmode shape.

N

Appendix: Tables of Phase 0 optimization setup Pareto front and Pareto set

Table N.1: Pareto front of Phase 0 optimization setup.

No.	1^{st} Eig.freq. [Hz]	5^{th} Eig.freq. [Hz]
1	15.18	157.40
2	15.20	156.33
3	15.24	153.89
4	15.18	157.59
5	15.12	165.16
6	15.30	149.84
7	15.18	156.50
8	15.35	149.77
9	15.29	150.49
10	15.25	152.64
11	15.25	151.50
12	15.18	157.40
13	15.24	153.89
14	15.27	151.24
15	15.16	158.01
16	15.15	160.83
17	15.15	160.17
18	15.14	162.44
19	15.26	151.42
20	15.14	161.74
21	15.25	152.81
22	15.18	157.59
23	15.39	149.70
24	15.29	150.49
25	15.09	165.31

Table N.2: Pareto set of Phase 0 optimization setup.

No.	L [mm]	L_2 [mm]	L_3 [mm]	α [°]	α_3 [°]	w [mm]	w_2 [mm]	w_3 [mm]
1	28.11	24.24	27.70	42.00	51.40	2.670	2.245	2.632
2	28.13	24.23	27.67	41.13	52.34	2.682	2.253	2.631
3	28.12	24.11	27.69	41.41	53.97	2.671	2.249	2.636
4	28.13	24.23	27.66	41.03	51.61	2.678	2.252	2.632
5	28.13	24.23	27.66	41.07	47.54	2.679	2.252	2.636
6	28.13	24.24	27.67	41.24	59.01	2.683	2.252	2.636
7	28.13	24.23	27.66	41.02	52.25	2.675	2.252	2.633
8	28.13	24.27	27.71	42.20	58.94	2.697	2.252	2.631
9	28.13	24.23	27.63	41.19	57.91	2.677	2.252	2.635
10	28.13	24.24	27.64	41.10	55.14	2.684	2.248	2.634
11	28.13	24.25	27.64	41.41	56.23	2.677	2.252	2.635
12	28.11	24.24	27.70	42.00	51.40	2.671	2.248	2.632
13	28.12	24.11	27.69	41.41	53.97	2.671	2.249	2.636
14	28.13	24.24	27.65	41.17	56.55	2.684	2.251	2.635
15	28.13	24.23	27.65	41.02	51.38	2.671	2.250	2.633
16	28.13	24.23	27.67	41.07	49.71	2.681	2.252	2.636
17	28.13	24.19	27.67	41.23	50.08	2.671	2.250	2.636
18	28.13	24.23	27.65	40.93	48.97	2.670	2.252	2.636
19	28.13	24.23	27.63	41.08	56.50	2.675	2.252	2.635
20	28.14	24.23	27.66	40.88	49.34	2.677	2.252	2.632
21	28.12	24.14	27.68	41.39	54.89	2.675	2.249	2.635
22	28.13	24.23	27.66	41.03	51.61	2.677	2.252	2.632
23	28.13	24.27	27.72	41.60	59.43	2.716	2.253	2.631
24	28.13	24.23	27.63	41.19	57.91	2.677	2.252	2.635
25	28.13	24.23	27.66	40.07	47.54	2.679	2.252	2.636

O

Appendix: Tables of Phase 1 optimization setup Pareto fronts and Pareto sets

Table O.1: Phase 1 Pareto front of optimization with maximization of bandwidth and area above threshold. No proof mass. Population size 25, number of generations 15.

No.	OF1: BW	OF2: $\int \Sigma\sigma_{n,i} df \ \times 10^9$
1	83.78	0.9468
2	80.08	1.0354
3	79.90	1.0773
4	76.22	1.1853
5	78.22	1.1650
6	85.83	0.8858
7	82.35	1.0229
8	79.88	1.0802
9	83.27	0.9594
10	82.77	0.9796
11	86.43	0.7818
12	79.93	1.0680
13	73.68	1.2160
14	80.70	1.0265
15	85.02	0.8986

Table O.2: Phase 1 Pareto set of optimization with maximization of bandwidth and area above threshold. No proof mass. Population size 25, number of generations 15.

No.	L [mm]	L_2 [mm]	L_3 [mm]	α [°]	α_3 [°]	w [mm]	w_2 [mm]	w_3 [mm]
1	25.16	24.24	28.60	32.00	38.07	3.488	2.042	3.522
2	23.97	24.13	28.49	32.13	37.70	3.531	2.099	3.509
3	23.59	24.15	28.68	32.02	38.19	3.509	2.321	3.509
4	21.63	23.06	29.55	38.27	38.06	3.539	2.496	3.453
5	21.59	23.74	30.07	38.25	31.90	3.548	2.540	3.443
6	23.95	25.97	29.40	32.29	31.44	3.517	2.273	3.509
7	24.27	24.21	28.57	32.07	38.03	3.498	2.071	3.519
8	23.59	24.15	28.68	32.14	38.19	3.509	2.321	3.509
9	24.97	24.24	28.58	32.01	38.04	3.497	2.051	3.521
10	25.04	24.18	28.61	32.05	38.09	3.507	2.103	3.51
11	25.12	25.13	29.35	32.24	33.68	3.492	2.263	3.513
12	23.05	24.08	29.38	32.31	38.34	3.530	2.400	3.468
13	21.54	22.48	29.71	38.20	38.49	3.553	2.606	3.439
14	24.39	24.18	28.62	32.12	38.15	3.504	2.260	3.512
15	23.95	25.97	29.40	32.29	31.44	3.761	2.273	3.509

Table O.3: Phase 1 Pareto front of optimization with maximization of bandwidth and area above threshold, with added proof masses (PM). Population size 25, number of generations 15.

No.	OF1: BW	OF2: $\int \Sigma\sigma_{n,i}df \ \times 10^9$
1	87.27	0.8946
2	73.69	1.1998
3	83.56	0.9598
4	81.05	1.1679
5	82.13	0.9906
6	88.05	0.8436
7	89.99	0.8410
8	72.72	1.2195
9	90.49	0.7416
10	91.42	0.7294

Table O.4: Phase 1 Pareto set of optimization with maximization of bandwidth and area above threshold. With proof mass. Population size 25, number of generations 15.

No.	L [mm]	L_2 [mm]	L_3 [mm]	α [°]	α_3 [°]	w [mm]	w_2 [mm]	w_3 [mm]	PM ratio from end L_3 [%]	Mass PM [g]
1	23.87	25.23	25.77	51.16	51.63	3.061	1.906	1.848	18.86	0.2706
2	20.85	24.91	23.82	63.49	71.00	3.081	1.845	1.938	15.63	0.3875
3	22.41	25.13	25.21	51.16	51.58	3.210	1.889	1.853	16.53	0.2740
4	21.41	24.61	25.17	51.09	51.38	3.354	1.883	1.882	16.36	0.2787
5	21.65	24.82	25.25	51.08	51.20	3.267	1.885	1.895	16.64	0.2773
6	25.14	24.79	24.77	50.86	51.18	3.351	1.896	1.936	16.58	0.2523
7	25.84	24.83	23.35	50.83	50.70	3.337	1.929	1.990	16.78	0.2474
8	20.87	24.77	23.84	63.47	70.99	3.115	1.855	1.938	15.70	0.3762
9	26.36	24.72	25.78	50.77	52.07	3.236	1.972	1.850	17.35	0.2806
10	27.15	24.89	25.61	51.03	51.36	3.313	1.979	1.947	15.98	0.2600

Table O.5: Phase 1 Pareto front of optimization with maximization of bandwidth and area above threshold. Population size 30, number of generations 25.

No.	OF1: BW	OF2: $\int \Sigma\sigma_{n,i}df \ \times 10^9$
1	77.03	0.9334
2	79.75	0.8744
3	79.85	0.8609
4	80.16	0.8571
5	81.72	0.7604
6	79.92	0.8608
7	75.62	0.9532
8	77.86	0.8956
9	76.90	0.9339
10	75.78	0.9449
11	75.86	0.9373
12	73.65	0.9999
13	81.55	0.8008
14	74.71	0.9908
15	81.62	0.7985
16	69.96	1.0150
17	80.42	0.8451
18	76.45	0.9340
19	76.85	0.9339
20	78.77	0.8945
21	79.12	0.8864
22	70.11	1.0140
23	70.68	1.0120
24	77.07	0.9182
25	82.17	0.7548
26	70.89	1.0120
27	77.75	0.9153
28	75.55	0.9663
29	74.42	0.9934

Table O.6: Phase 1 Pareto set of optimization with maximization of bandwidth and area above threshold. No proof mass. Population size 30, number of generations 25.

No.	L [mm]	L_2 [mm]	L_3 [mm]	α [°]	α_3 [°]	w [mm]	w_2 [mm]	w_3 [mm]
1	26.82	28.08	23.43	51.27	64.20	2.653	2.154	2.514
2	28.37	27.72	23.76	51.36	63.98	2.639	2.082	2.445
3	28.17	28.02	23.75	51.17	63.97	2.661	2.164	2.514
4	28.17	28.02	23.75	51.17	64.22	2.661	2.164	2.514
5	28.65	28.34	24.05	45.19	64.57	2.718	2.114	2.664
6	28.17	28.02	23.75	51.17	64.03	2.661	2.164	2.514
7	27.35	27.94	23.13	51.27	63.76	2.669	2.175	2.440
8	27.61	27.56	23.66	51.35	64.03	2.653	2.097	2.569
9	27.01	28.00	23.74	51.28	63.74	2.660	2.179	2.431
10	27.46	27.94	23.13	51.23	63.75	2.669	2.167	2.440
11	26.96	27.91	23.40	51.17	64.15	2.662	2.193	2.514
12	26.51	27.90	22.88	51.16	63.98	2.668	2.193	2.476
13	28.82	28.05	23.73	51.11	64.42	2.668	2.184	2.601
14	26.58	27.95	23.04	51.20	63.85	2.665	2.187	2.457
15	28.82	28.05	23.73	51.11	64.48	2.668	2.184	2.601
16	26.55	27.93	22.48	51.18	63.97	2.667	2.187	2.458
17	28.32	28.04	23.74	51.26	64.27	2.665	2.184	2.563
18	27.48	28.01	23.08	51.17	64.01	2.663	2.182	2.482
19	27.71	27.83	23.18	51.33	63.98	2.641	2.156	2.441
20	27.57	27.95	23.67	51.17	64.03	2.665	2.171	2.515
21	27.86	27.93	23.71	51.20	63.98	2.646	2.129	2.471
22	26.56	27.93	22.50	51.18	63.97	2.668	2.186	2.459
23	26.55	27.92	22.56	51.18	63.96	2.668	2.187	2.459
24	27.28	27.91	23.50	51.17	64.14	2.662	2.180	2.515
25	28.65	28.34	24.05	44.88	64.82	2.718	2.114	2.664
26	26.56	27.92	22.58	51.19	63.96	2.669	2.183	2.460
27	27.18	27.95	23.55	51.17	64.14	2.661	2.169	2.514
28	27.05	27.90	23.06	51.20	63.98	2.667	2.190	2.476
29	26.55	27.94	23.00	51.19	63.87	2.668	2.192	2.467

Table O.7: Phase 1 Pareto front of optimization with maximization of bandwidth and area above threshold, with added proof masses. Population size 30, number of generations 25.

No.	OF1: BW	OF2: $\int \Sigma \sigma_{n,i} df \ \times 10^9$
1	87.07	1.3210
2	96.14	1.1520
3	96.41	0.6104
4	91.55	1.1820
5	90.11	1.1850

Table O.8: Phase 1 Pareto set of optimization with maximization of bandwidth and area above threshold. With proof mass. Population size 30, number of generations 25.

No.	L [mm]	L_2 [mm]	L_3 [mm]	α [°]	α_3 [°]	w [mm]	w_2 [mm]	w_3 [mm]	PM ratio from end L_3 [%]	Mass PM [g]
1	25.73	26.00	24.82	69.71	57.08	2.875	3.405	1.667	12.73	0.1022
2	27.00	25.92	24.84	69.69	57.31	2.875	2.943	1.600	11.66	0.1508
3	26.78	26.99	25.42	64.63	57.52	2.854	2.174	1.780	10.94	0.1342
4	25.67	25.97	24.82	69.71	57.08	2.875	2.893	1.675	12.34	0.1270
5	25.64	25.93	24.84	69.71	57.34	2.875	3.371	1.725	11.15	0.1292

Table O.9: Phase 1 Pareto front of optimization with maximization of bandwidth and area above threshold. Population size 45, number of generations 20.

No.	OF1: BW	OF2: $\int \Sigma\sigma_{n,i}df \ \times 10^9$
1	72.37	1.385
2	83.52	1.206
3	77.74	1.326
4	79.92	1.303
5	83.33	1.226
6	75.12	1.365
7	85.41	1.131
8	81.52	1.253
9	81.27	1.276
10	76.76	1.352
11	84.67	1.171
12	80.70	1.288
13	81.32	1.269
14	77.56	1.327
15	89.31	1.032
16	81.53	1.246
17	79.66	1.316
18	80.88	1.286
19	82.99	1.236
20	79.68	1.309
21	82.34	1.239
22	88.81	1.130
23	83.84	1.204
24	84.31	1.199
25	82.03	1.240
26	81.21	1.282
27	80.54	1.298
28	85.39	1.163
29	72.36	1.414
30	76.94	1.351
31	79.68	1.314
32	81.32	1.261
33	77.85	1.322
34	93.27	1.020

Table O.10: Phase 1 Pareto set of optimization with maximization of bandwidth and area above threshold. No proof mass. Population size 45, number of generations 20.

No.	L [mm]	L_2 [mm]	L_3 [mm]	α [°]	α_3 [°]	w [mm]	w_2 [mm]	w_3 [mm]
1	20.71	17.15	32.49	51.35	63.73	2.959	2.210	3.160
2	23.71	17.33	32.13	51.30	64.07	2.951	2.107	3.165
3	22.87	17.22	31.69	51.11	64.10	3.002	2.225	3.082
4	22.65	17.22	32.01	51.32	63.99	2.972	2.214	3.152
5	23.69	17.15	32.20	51.30	63.76	3.002	2.079	3.150
6	21.95	17.07	31.65	51.09	63.62	3.033	2.158	3.087
7	24.85	17.30	32.28	51.15	63.73	3.042	2.016	3.124
8	23.69	17.28	31.95	51.09	63.84	3.023	2.112	3.099
9	24.12	16.93	31.19	51.10	63.71	3.044	2.074	3.054
10	22.09	17.11	32.01	51.17	63.74	3.031	2.275	3.153
11	24.33	17.30	32.17	51.15	63.77	3.043	2.016	3.121
12	23.16	17.17	32.08	51.12	63.80	3.035	2.062	3.107
13	23.42	17.28	32.12	51.12	63.78	3.049	2.179	3.111
14	22.87	17.22	31.69	51.11	63.85	3.002	2.225	3.082
15	25.60	17.27	31.12	50.95	59.21	3.084	1.737	3.199
16	23.67	17.28	32.14	51.10	63.82	3.037	2.095	3.109
17	22.71	17.19	32.02	51.14	63.81	3.032	2.089	3.118
18	22.81	17.17	32.15	51.30	63.83	2.976	2.191	3.143
19	23.65	17.25	32.25	51.16	63.76	3.032	2.124	3.127
20	22.54	17.20	32.04	51.32	63.84	2.973	2.214	3.139
21	23.59	17.28	32.16	51.10	63.80	3.045	2.025	3.109
22	25.51	17.03	31.17	51.14	62.94	3.060	1.746	2.995
23	24.04	17.26	32.34	51.18	63.81	3.004	2.125	3.102
24	24.11	17.26	32.35	51.17	63.79	3.028	2.102	3.123
25	23.55	17.33	31.99	51.26	64.03	2.991	2.146	3.170
26	22.82	17.27	32.14	51.29	63.88	2.981	2.183	3.150
27	22.91	17.05	32.06	51.16	63.75	3.020	2.048	3.151
28	24.44	17.28	32.24	51.29	63.90	3.005	2.041	3.168
29	20.71	17.15	32.49	51.10	63.73	2.959	2.210	3.160
30	22.06	17.11	31.75	51.09	63.66	3.033	2.089	3.117
31	22.84	17.22	31.98	51.12	63.85	2.986	2.217	3.131
32	23.52	17.26	32.04	51.12	63.78	3.025	2.132	3.084
33	22.62	17.22	31.81	51.15	63.83	3.005	2.202	3.196
34	25.55	18.83	31.39	51.38	63.74	3.084	1.735	3.209

Table O.11: Phase 1 Pareto front of optimization with maximization of bandwidth and area above threshold. With added proof masses. Population size 45, number of generations 20.

No.	OF1: BW	OF2: $\int \Sigma\sigma_{n,i}df$ $\times 10^9$
1	77.67	1.201
2	71.60	1.712
3	86.20	1.143
4	74.93	1.220
5	85.47	1.150
6	77.40	1.205
7	75.61	1.216
8	83.71	1.168
9	75.91	1.207
10	84.67	1.156
11	75.71	1.215
12	87.21	1.121
13	77.66	1.204
14	89.92	1.113
15	82.22	1.177
16	85.66	1.144
17	74.43	1.620
18	74.23	1.647

Table O.12: Phase 1 Pareto set of optimization with maximization of bandwidth and area above threshold. With proof mass. Population size 45, number of generations 20.

No.	L [mm]	L_2 [mm]	L_3 [mm]	α [°]	α_3 [°]	w [mm]	w_2 [mm]	w_3 [mm]	PM ratio from end L_3 [%]	Mass PM [g]
1	21.76	25.25	21.88	51.09	44.41	3.149	2.406	1.993	6.227	0.2273
2	22.65	26.20	20.57	57.42	45.41	2.934	3.380	1.919	25.41	0.2385
3	21.69	25.25	25.29	50.97	44.36	3.144	2.399	1.992	17.66	0.2219
4	21.73	25.25	22.29	50.90	44.36	3.149	2.403	1.991	15.21	0.2173
5	21.64	25.25	25.15	50.96	44.36	3.140	2.400	1.993	17.42	0.2228
6	21.76	25.25	21.88	51.09	44.35	3.149	2.406	1.993	7.062	0.2273
7	21.66	25.25	22.42	51.09	44.45	3.136	2.396	1.993	18.11	0.2264
8	21.67	25.25	24.60	50.99	44.37	3.146	2.401	1.992	16.83	0.2213
9	21.73	25.25	22.62	50.94	44.36	3.145	2.403	1.991	17.28	0.2212
10	21.65	25.25	24.97	50.97	44.36	3.137	2.400	1.993	17.52	0.2253
11	21.67	25.25	22.68	50.94	44.36	3.141	2.403	1.991	18.40	0.2232
12	21.65	25.25	25.14	51.08	44.36	3.137	2.401	1.992	13.87	0.2256
13	21.76	25.25	21.88	51.09	44.40	3.149	2.406	1.993	6.743	0.2273
14	21.70	25.25	25.28	51.09	44.38	3.135	2.402	1.993	10.83	0.2268
15	21.66	25.25	24.37	51.05	44.38	3.133	2.397	1.993	17.83	0.2248
16	21.64	25.25	25.04	51.08	44.37	3.139	2.398	1.993	16.03	0.2234
17	22.65	26.20	20.57	57.42	45.41	2.934	3.380	1.919	12.91	0.2385
18	22.65	26.20	20.57	57.42	45.28	2.934	3.380	1.919	12.91	0.2385

P

Appendix: Tables of Phase 2 optimization setup Pareto fronts and Pareto sets

Table P.1: Phase 2 Pareto front and Pareto set of optimization with maximization of bandwidth (OF1: BW) and area under FRF (OF2: $\int \Sigma\sigma_{n,i}A_i df$). With proof mass. Population size 25, number of generations 15.

	OF1	OF2	ub1	ub2	ub3	ub4	ub5	ub6	ub7
No.	BW [Hz]	$\int \Sigma\sigma_{n,i}A_i df$	[%]	[%]	[%]	[%]	[%]	[%]	[%]
1	78.11	72270	43.11	74.33	54.44	78.29	54.53	40.10	53.54
2	78.12	68430	56.36	49.80	41.14	67.70	81.43	88.40	51.41
3	78.53	53230	28.07	35.02	33.12	75.98	44.60	64.16	39.20
4	77.74	92720	57.74	87.92	87.57	80.59	81.45	84.11	69.75
5	77.76	86500	65.88	76.95	80.31	80.00	75.39	78.12	55.13
6	77.76	86770	56.38	82.99	73.20	68.80	70.50	86.82	69.87
7	78.10	78090	33.15	76.05	74.55	83.96	74.48	43.24	54.25
8	77.75	91190	59.17	83.72	86.34	80.76	74.11	82.26	73.29
9	78.52	53760	28.59	41.69	35.73	86.70	46.22	35.94	40.10
10	71.61	93050	72.13	87.89	88.13	81.57	83.21	89.06	67.63
11	77.75	90070	59.98	83.29	84.92	80.55	81.02	83.38	59.84
12	77.74	91750	67.73	83.89	87.51	81.52	77.63	85.28	72.76
13	71.61	93110	70.12	87.92	87.78	81.14	81.96	86.44	69.01

Table P.2: Phase 2 Pareto front and Pareto set of optimization with maximization of bandwidth (OF1: BW) and area under FRF (OF2: $\int \Sigma\sigma_{n,i}A_i df$). With proof mass. Population size 30, number of generations 25.

No.	OF1 BW [Hz]	OF2 $\int \Sigma\sigma_{n,i}A_i df$	ub1 [%]	ub2 [%]	ub3 [%]	ub4 [%]	ub5 [%]	ub6 [%]	ub7 [%]
1	78.12	77400	42.47	58.94	78.28	77.75	50.30	58.22	73.42
2	78.11	83340	34.16	72.28	82.75	79.72	80.94	78.61	59.15
3	77.76	96920	62.19	99.63	90.04	91.78	76.04	85.50	71.65
4	77.77	88090	57.73	83.00	83.10	78.67	65.74	57.31	81.56
5	77.75	97440	62.29	99.63	91.57	91.34	80.47	85.58	71.08
6	77.75	97090	62.42	99.63	90.34	91.56	76.80	85.76	71.33
7	77.76	96440	62.07	99.63	86.97	88.40	79.58	84.71	75.51
8	77.77	87360	60.13	80.43	82.62	78.10	71.61	57.26	81.38
9	78.55	54350	32.69	39.70	32.40	48.53	39.70	58.63	73.11
10	78.54	55820	32.55	37.15	32.24	48.08	89.32	58.67	72.47

Table P.3: Phase 2 Pareto front and Pareto set of optimization with maximization of area above threshold (OF1: $\int \Sigma\sigma_{n,i}df$) and area under FRF (OF2: $\int \Sigma\sigma_{n,i}A_i df$). With proof mass. Population size 25, number of generations 15.

No.	OF1 $\int \Sigma\sigma_{n,i}df$	OF2 $\int \Sigma\sigma_{n,i}A_i df$	ub1 [%]	ub2 [%]	ub3 [%]	ub4 [%]	ub5 [%]	ub6 [%]	ub7 [%]
1	2.358	93130	58.87	99.39	80.36	80.28	88.71	70.92	81.23
2	2.711	72740	32.17	76.98	80.12	21.79	61.07	53.96	54.05
3	2.681	76350	31.97	77.79	80.35	37.49	59.29	65.42	54.36
4	2.424	89440	58.87	99.39	80.36	80.28	88.71	70.92	31.23
5	2.682	76140	30.98	76.96	78.97	48.65	57.01	56.08	57.62
6	2.375	90090	58.14	83.51	81.98	80.20	81.12	73.66	80.88
7	2.693	75520	32.03	77.48	80.33	38.54	59.52	56.66	54.30
8	2.417	90030	58.87	99.39	80.36	80.28	88.71	70.92	37.89
9	2.615	82370	30.21	77.26	98.83	79.07	32.08	78.03	41.35
10	2.686	75610	32.17	76.98	92.62	21.79	86.07	66.46	29.05
11	2.487	89440	44.51	92.69	80.31	79.41	79.54	71.21	65.42
12	2.564	85660	37.11	92.20	80.22	79.13	50.82	71.33	54.03
13	2.596	83660	36.17	85.91	80.36	67.67	62.46	68.37	56.81
14	2.392	90060	48.79	89.50	81.27	83.85	81.48	94.38	52.34
15	2.368	92070	58.37	92.13	81.17	80.27	88.15	71.85	81.11
16	2.657	79760	30.93	76.97	97.38	62.90	33.24	59.34	52.39
17	2.359	92130	59.13	93.96	81.22	80.28	86.18	71.09	81.17
18	2.675	78790	35.23	88.18	80.18	36.39	63.49	55.14	55.07
19	2.678	76830	31.63	76.87	79.82	45.33	64.53	62.05	52.76
20	2.711	74860	31.55	76.85	79.04	46.97	41.55	54.28	53.82
21	2.610	82780	32.68	86.26	80.17	72.11	60.75	66.06	53.91
22	2.651	80920	38.43	92.82	80.29	37.29	70.10	55.54	55.18

Table P.4: Phase 2 Pareto front and Pareto set of optimization with maximization of area above threshold (OF1: $\int \Sigma\sigma_{n,i}df$) and area under FRF (OF2: $\int \Sigma\sigma_{n,i}A_i df$). With proof mass. Population size 30, number of generations 25. Run 1.

No.	OF1 $\int \Sigma\sigma_{n,i}df$	OF2 $\int \Sigma\sigma_{n,i}A_i df$	ub1 [%]	ub2 [%]	ub3 [%]	ub4 [%]	ub5 [%]	ub6 [%]	ub7 [%]
1	2.609	78860	33.05	93.09	89.55	58.45	36.87	37.93	61.41
2	2.453	86940	40.67	93.44	95.92	73.98	57.41	50.72	85.47
3	2.421	88440	44.55	93.75	96.44	73.00	71.38	50.17	87.53
4	2.581	80690	33.30	93.21	89.56	59.69	47.82	44.43	62.35
5	2.590	79940	32.68	92.47	90.09	54.94	42.59	48.10	63.45
6	2.525	83980	33.94	93.32	92.42	71.14	65.21	50.75	62.54
7	2.306	92340	56.57	94.53	97.44	75.06	71.34	74.07	87.53
8	2.335	90660	53.61	93.93	97.40	73.08	71.76	64.25	87.51
9	2.634	77010	32.80	91.77	89.94	41.44	42.44	38.06	61.57
10	2.502	84300	34.60	93.29	91.98	66.26	69.59	48.09	81.71
11	2.571	81460	33.31	92.68	89.72	59.89	50.54	49.65	62.73
12	2.571	81730	32.75	92.38	93.17	61.04	48.18	48.75	63.09
13	2.602	79180	33.07	92.67	92.41	59.06	37.31	37.44	61.56
14	2.393	89190	49.66	93.56	97.25	73.36	65.06	59.45	77.34
15	2.357	90220	53.15	93.48	97.28	71.45	70.33	63.77	83.03
16	2.615	77860	33.17	92.39	89.58	42.98	42.52	41.97	61.74
17	2.499	84500	37.84	92.89	94.43	68.67	56.32	51.45	68.94
18	2.330	91230	52.51	94.45	97.36	74.32	66.39	72.57	86.88
19	2.543	82230	33.53	92.73	90.02	68.48	52.40	50.53	62.63
20	2.497	86240	33.23	99.13	98.79	64.77	55.07	64.54	71.79
21	2.384	89900	45.63	94.53	91.19	75.06	58.84	74.07	87.53
22	2.541	83880	33.09	97.36	92.22	61.96	49.68	57.57	68.93
23	2.497	84770	36.44	93.44	96.46	71.16	67.51	49.38	62.50
24	2.627	77660	32.83	92.29	89.88	46.28	42.21	37.96	61.42
25	2.409	88540	45.71	94.21	92.80	72.84	58.60	61.25	86.51
26	2.483	86670	34.21	97.14	95.54	68.45	58.04	61.20	82.20
27	2.450	88010	34.78	93.28	96.94	73.15	67.61	70.36	80.16
28	2.542	83090	33.31	92.68	89.72	59.89	75.54	49.65	68.98

Table P.5: Phase 2 Pareto front and Pareto set of optimization with maximization of area above threshold (OF1: $\int \Sigma\sigma_{n,i}df$) and area under FRF (OF2: $\int \Sigma\sigma_{n,i}A_i df$). With proof mass. Population size 30, number of generations 25. Run 2.

	OF1	OF2	ub1	ub2	ub3	ub4	ub5	ub6	ub7
No.	$\int \Sigma\sigma_{n,i}df$	$\int \Sigma\sigma_{n,i}A_i df$	[%]	[%]	[%]	[%]	[%]	[%]	[%]
1	2.568	79870	35.35	85.74	94.66	73.34	34.56	51.73	39.59
2	2.505	83000	35.51	85.54	93.91	81.86	50.92	56.93	54.54
3	2.397	87280	45.17	84.45	94.55	87.08	74.25	73.10	50.74
4	2.223	90410	65.79	85.60	95.80	92.45	92.56	81.61	51.77
5	2.538	82150	35.46	88.22	94.27	79.10	43.39	53.42	45.96
6	2.650	70880	32.37	77.36	85.81	65.70	22.38	30.90	33.84
7	2.650	70760	32.49	78.77	82.15	67.19	25.94	30.62	32.30
8	2.618	76490	33.77	91.70	86.43	70.05	30.33	35.07	37.49
9	2.335	89640	46.11	85.62	95.61	92.39	90.76	77.99	60.16
10	2.648	73040	33.24	84.69	85.17	63.39	29.73	30.54	32.67
11	2.671	66680	32.13	70.19	77.66	64.82	22.04	28.54	32.07
12	2.508	82270	41.36	88.75	87.23	72.72	36.11	69.16	40.31
13	2.594	79020	34.10	86.99	94.46	67.15	33.01	50.00	38.68
14	2.361	88870	46.36	85.62	95.13	91.15	91.97	75.39	50.21
15	2.666	68680	28.42	72.23	84.52	65.82	25.63	29.19	39.18
16	2.461	85640	40.63	88.27	93.58	86.81	54.37	65.52	47.92
17	2.662	69690	29.38	74.31	85.80	65.78	25.48	30.05	37.06
18	2.575	79440	34.90	91.33	87.93	72.94	33.65	50.71	38.12
19	2.538	81310	35.42	85.58	93.97	80.27	44.88	52.09	44.12
20	2.611	77530	34.62	93.26	87.60	70.63	31.89	35.07	37.49
21	2.601	77990	34.46	87.56	93.63	66.54	30.60	45.27	36.95
22	2.430	86510	40.39	86.48	94.29	87.74	77.15	69.72	46.40
23	2.495	83850	35.06	85.48	90.16	90.24	62.77	55.60	55.82
24	2.632	74100	34.46	84.91	87.30	64.10	31.72	31.11	34.57
25	2.630	75230	33.51	87.54	86.03	65.97	30.33	36.36	36.48
26	2.650	70200	32.70	77.35	83.46	65.45	22.83	28.72	34.75
27	2.475	84950	37.55	85.56	94.28	83.53	63.25	63.28	53.94
28	2.617	77200	33.77	91.70	89.55	70.05	30.33	35.07	37.49
29	2.395	88120	44.61	86.04	94.65	89.36	76.52	74.89	49.69
30	2.641	73360	33.24	84.69	85.17	63.39	32.66	30.54	35.79

Table P.6: Phase 2 Pareto front and Pareto set of optimization with maximization of bandwidth (OF1: BW) and area under FRF (OF2: $\int \Sigma\sigma_{n,i}A_i df$). No proof mass. Population size 25, number of generations 15.

	OF1	OF2	ub1	lb2	lb3	ub4	ub5	ub6	ub7
No.	BW [Hz]	$\int \Sigma\sigma_{n,i}A_i df$	[%]	[%]	[%]	[%]	[%]	[%]	[%]
1	79.00	43920	31.98	44.18	72.12	75.81	28.65	31.59	29.26
2	72.49	61450	56.56	73.50	96.59	93.06	59.62	81.66	55.53
3	78.98	44190	32.71	45.54	63.54	58.83	35.58	31.46	27.27
4	72.53	58890	45.73	71.53	97.50	88.98	60.29	47.90	46.99
5	78.49	53150	31.81	56.01	95.82	34.72	68.66	63.13	33.69
6	72.52	59680	51.34	69.58	97.55	76.82	64.32	80.03	41.22
7	78.38	54250	32.86	79.44	69.27	50.94	58.00	63.23	36.46
8	78.52	48180	31.20	30.79	97.69	49.08	62.31	34.37	32.60
9	72.47	63530	57.71	95.54	88.54	58.83	85.58	56.46	52.27
10	78.40	53590	32.31	70.02	72.44	38.54	60.82	75.94	38.17
11	78.37	54340	32.34	78.54	75.86	36.14	68.83	75.64	30.20
12	73.31	57320	44.27	76.87	94.69	50.97	58.70	68.93	26.74
13	78.37	55080	34.73	79.14	77.13	40.42	67.06	70.99	29.53
14	79.02	43810	31.91	33.79	72.83	75.73	30.49	41.81	31.45
15	72.51	60900	53.83	73.02	97.18	77.62	62.88	80.55	47.82

Table P.7: Phase 2 Pareto front and Pareto set of optimization with maximization of bandwidth (OF1: BW) and area under FRF (OF2: $\int \Sigma\sigma_{n,i}A_i df$). No proof mass. Population size 30, number of generations 25.

	OF1	OF2	ub1	lb2	lb3	ub4	ub5	ub6	ub7
No.	BW [Hz]	$\int \Sigma\sigma_{n,i}A_i df$	[%]	[%]	[%]	[%]	[%]	[%]	[%]
1	72.49	64640	65.55	88.47	95.47	93.81	78.82	87.50	59.90
2	72.57	62130	40.01	87.92	98.58	93.87	86.15	97.74	63.72
3	77.58	56180	29.14	76.68	94.24	49.24	69.31	47.76	44.22
4	72.45	65030	60.64	90.67	94.04	91.11	80.32	88.08	61.80
5	78.37	55990	31.95	79.16	90.09	49.38	56.83	70.14	36.98
6	72.41	65110	65.29	92.03	93.09	86.88	78.18	88.04	62.81
7	72.48	64960	62.88	90.03	94.20	93.24	79.44	87.12	61.62
8	72.48	64680	63.54	87.36	95.70	90.52	79.23	88.05	62.01
9	72.54	63670	48.19	87.02	97.92	93.92	81.86	90.41	60.36
10	78.44	55300	34.55	85.76	95.22	40.42	49.60	68.50	27.19
11	73.22	57730	42.11	78.77	88.29	46.35	66.89	91.51	28.18
12	78.39	55510	37.26	81.00	79.61	50.25	56.10	81.78	28.48
13	72.51	64380	54.95	87.17	96.83	93.41	79.62	87.93	60.54
14	78.46	53690	30.08	72.53	95.18	43.87	57.32	52.96	30.85
15	77.54	56740	30.40	76.68	92.86	50.43	67.55	60.42	43.46
16	72.50	64580	64.64	91.85	96.39	94.01	79.28	88.69	55.47

Table P.8: Phase 2 Pareto front and Pareto set of optimization with maximization of area above threshold (OF1: $\int \Sigma\sigma_{n,i}df$) and area under FRF (OF2: $\int \Sigma\sigma_{n,i}A_idf$). No proof mass. Population size 25, number of generations 15.

	OF1	OF2	ub1	lb2	lb3	ub4	ub5	ub6	ub7
No.	$\int \Sigma\sigma_{n,i}df$	$\int \Sigma\sigma_{n,i}A_idf$	[%]	[%]	[%]	[%]	[%]	[%]	[%]
1	1.703	55580	34.32	72.65	84.58	67.58	52.80	46.64	48.13
2	1.809	48880	24.63	77.90	81.80	83.94	29.90	37.09	40.86
3	1.508	61130	57.78	67.36	86.01	80.32	64.09	54.43	81.23
4	1.774	51550	27.25	79.14	81.05	85.47	36.37	43.53	43.73
5	1.679	56730	33.99	87.12	95.59	95.13	42.50	43.34	49.31
6	1.465	62610	57.78	69.46	86.01	81.62	82.77	67.34	81.23
7	1.729	54670	32.82	76.30	82.99	67.82	47.70	46.55	44.16
8	1.767	52190	28.83	76.15	80.33	83.10	39.14	44.42	44.06
9	1.660	58360	30.13	76.44	89.89	85.52	67.22	71.59	58.08
10	1.821	47110	26.88	77.57	59.81	85.63	28.76	36.55	40.06
11	1.660	57310	39.36	76.25	82.70	87.93	57.25	29.45	57.21
12	1.743	53890	28.83	84.78	91.14	88.03	38.38	43.51	43.97
13	1.681	55910	38.72	83.59	90.44	86.74	39.70	57.59	35.83
14	1.403	63570	64.03	71.27	91.09	91.26	85.96	86.46	81.23
15	1.749	53040	30.88	81.81	87.27	83.94	35.37	39.44	43.60
16	1.611	59930	39.08	71.42	89.17	83.87	64.12	54.30	76.75
17	1.837	44110	25.39	52.69	63.78	90.71	31.59	29.41	41.00
18	1.799	50060	24.63	77.90	83.41	83.94	35.50	37.09	40.86
19	1.851	42910	25.30	52.21	60.53	91.23	29.16	24.00	41.04
20	1.821	47110	26.88	77.57	59.81	85.63	28.76	36.55	40.06
21	1.828	46430	24.76	78.43	63.83	87.02	26.24	36.42	40.57
22	1.777	51000	27.61	78.83	73.29	85.31	37.37	41.66	44.44
23	1.846	42920	24.04	41.98	69.85	97.43	33.77	23.73	42.54
24	1.792	50490	26.00	79.96	83.17	85.50	33.03	37.68	42.43
25	1.645	59110	31.30	77.12	92.24	82.40	68.20	71.59	60.62

Table P.9: Phase 2 Pareto front and Pareto set of optimization with maximization of area above threshold (OF1: $\int \Sigma\sigma_{n,i}df$) and area under FRF (OF2: $\int \Sigma\sigma_{n,i}A_i df$). No proof mass. Population size 30, number of generations 25.

	OF1	OF2	ub1	lb2	lb3	ub4	ub5	ub6	ub7
No.	$\int \Sigma\sigma_{n,i}df$	$\int \Sigma\sigma_{n,i}A_i df$	[%]	[%]	[%]	[%]	[%]	[%]	[%]
1	1.786	50820	27.62	67.69	72.10	51.10	40.17	51.49	53.38
2	1.548	62750	43.58	91.35	78.53	82.58	78.24	71.35	68.21
3	1.647	59060	38.02	82.59	77.91	50.63	63.21	59.16	62.30
4	1.655	58350	27.21	90.34	74.86	83.60	70.85	62.90	74.86
5	1.862	42310	26.37	62.82	65.69	63.49	23.06	28.52	27.08
6	1.855	43960	26.37	63.38	65.18	42.69	24.76	28.18	42.73
7	1.835	45480	26.57	62.86	68.19	56.06	25.95	43.75	40.08
8	1.722	55380	29.52	83.14	75.51	60.67	52.90	50.02	55.96
9	1.671	57420	28.63	79.01	78.24	51.38	76.76	62.27	66.76
10	1.745	54010	29.04	74.73	78.94	54.64	46.34	56.91	55.42
11	1.742	54740	29.21	89.67	72.21	54.07	42.49	58.00	57.43
12	1.618	60190	38.69	85.43	78.44	72.20	66.81	59.50	63.86
13	1.817	47710	27.34	66.10	67.16	50.86	30.98	43.38	48.64
14	1.783	51300	26.68	80.98	67.62	63.40	40.24	49.40	47.52
15	1.758	52950	27.48	81.82	74.59	53.61	39.74	57.16	56.50
16	1.774	52000	27.95	83.74	68.15	51.07	42.42	50.30	45.90
17	1.545	63010	43.64	90.36	81.90	75.70	79.07	61.10	76.57
18	1.801	48770	26.63	64.28	69.54	68.22	36.62	54.93	42.87
19	1.746	53540	29.02	76.09	73.92	57.61	45.49	56.24	55.39
20	1.474	63940	48.93	95.85	78.10	84.48	82.98	71.11	81.41
21	1.800	49660	27.46	72.86	66.44	49.54	41.35	39.49	44.94
22	1.790	50270	28.14	70.63	73.34	60.95	41.16	43.33	40.42
23	1.769	52720	26.99	88.30	68.33	56.25	40.35	55.97	51.00
24	1.830	46860	26.50	65.56	66.86	49.03	29.27	46.20	45.36
25	1.831	46040	26.53	66.22	66.41	44.42	25.09	50.97	45.13
26	1.822	47220	27.17	65.93	67.05	49.77	29.52	46.07	46.16
27	1.849	44400	26.37	62.87	65.23	50.55	23.68	35.39	42.88
28	1.591	60970	36.43	86.48	78.10	71.98	82.98	71.11	68.91
29	1.586	61640	35.02	90.34	85.80	83.60	70.85	62.90	87.36
30	1.426	64390	56.74	95.85	78.10	96.98	82.98	71.11	81.41

DEPARTMENT OF MECHANICS AND MARITIME SCIENCES
CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden
www.chalmers.se

CHALMERS
UNIVERSITY OF TECHNOLOGY